

BA_PM.R

liliy

2022-01-08

#Исследование случай-контроль: бронхиальная астма и РМ
#Л.М.Фатхутдинова
#1 декабря 2021 г.

"Загрузка пакетов"

```
## [1] "Загрузка пакетов"
```

```
library(MASS)
library(knitr)
library(epiR)
```

```
## Загрузка требуемого пакета: survival
```

```
## Package epiR 2.0.40 is loaded
```

```
## Type help(epi.about) for summary information
```

```
## Type browseVignettes(package = 'epiR') to learn how to use epiR for applied  
epidemiological analyses
```

```
##
```

```
library(ggplot2)
library(oddsratio)
```

```
theme_set(theme_bw())
Sys.setenv(LANG = "en")
```

"Загрузка файла с данными + список переменных"

```
## [1] "Загрузка файла с данными + список переменных"
```

```
ba<-read.csv(file="ba9.csv",sep=";",dec=',')
names<-names(ba)
str(ba)
```

```
## 'data.frame':   141 obs. of  105 variables:
## $ ID           : chr  "3" "60" "74" "52" ...
## $ Kazan        : int   0 0 1 0 1 0 0 1 1 0 ...
## $ Point        : int   NA NA 1 NA 10 NA NA 13 1 NA ...
## $ TSPAver      : num   NA NA 0.164 NA 0.166 NA NA 0.15 0.164 NA ...
## $ TSPMaxAver   : num   NA NA 0.595 NA 0.628 ...
## $ TSPMaxMax    : num   NA NA 0.734 NA 0.769 NA NA 0.72 0.734 NA ...
## $ PM10Aver     : num   NA NA 0.094 NA 0.089 NA NA 0.081 0.094 NA ...
## $ PM10MaxAver  : num   NA NA 0.348 NA 0.38 ...
```

```

## $ PM10MaxMax      : num  NA NA 0.44 NA 0.45 NA NA 0.42 0.44 NA ...
## $ PM2.5Avr        : num  NA NA 0.031 NA 0.029 NA NA 0.025 0.031 NA ...
## $ PM2.5MaxAvr     : num  NA NA 0.172 NA 0.199 ...
## $ PM2.5MaxMax     : num  NA NA 0.253 NA 0.264 NA NA 0.379 0.253 NA ...
## $ TSPDT           : num  NA NA 0.038 NA 0.061 NA NA 0.058 0.038 NA ...
## $ PM10DT          : num  NA NA 0.03 NA 0.058 NA NA 0.046 0.03 NA ...
## $ PM25DT          : num  NA NA 0.028 NA 0.021 NA NA 0.026 0.028 NA ...
## $ Cu              : num  NA NA 0.56 NA 0.76 NA NA NA 0.56 NA ...
## $ C               : num  NA NA 93.4 NA 90.4 ...
## $ LAL1            : num  NA NA 0.0277 NA 0.0276 NA NA 0.0064 0.0277 NA ...
## $ LAL2            : num  NA NA 0.0694 NA 0.0553 NA NA 0.0382 0.0694 NA ...
## $ TBPLogMax       : num  NA NA 0.0221 NA 0.023 ...
## $ PLogMax         : num  NA NA 0.0126 NA 0.0129 ...
## $ TBLogMax        : num  NA NA 0.00948 NA 0.0101 ...
## $ TBPLogAvr       : num  NA NA 0.00804 NA 0.00794 ...
## $ PLogAvr         : num  NA NA 0.00457 NA 0.00447 ...
## $ TBLogAvr        : num  NA NA 0.00346 NA 0.00346 ...
## $ BA              : int   1 1 1 1 1 1 1 1 1 1 ...
## $ BAcid           : chr   "j45.0" "j45.1" "j45.1" "j45.0" ...
## $ BMI             : num   27.5 39 39.1 25.9 29.4 ...
## $ BMICd           : int    2 3 3 2 2 1 2 1 2 2 ...
## $ BAsevere        : int    2 3 2 2 2 3 3 1 2 2 ...
## $ BAobstr         : int    1 0 1 0 0 1 1 0 0 0 ...
## $ BAcontrol       : int    2 2 2 NA 2 NA 3 1 3 2 ...
## $ BAdebut         : int    1 1 0 1 0 0 0 0 0 0 ...
## $ IgE             : num   155 72.9 11.3 NA NA ...
## $ Eos             : num   276 291 286 510 321 ...
## $ Atopia          : int    1 0 0 1 1 0 1 1 0 0 ...
## $ Heredity        : int    1 0 1 0 1 0 0 0 0 1 ...
## $ Age             : num   39.7 56.7 37.7 38.8 46.5 ...
## $ Sex             : int    1 0 0 0 0 1 1 0 0 0 ...
## $ Family          : int    6 6 6 6 1 6 6 1 2 6 ...
## $ FamType         : int    3 3 3 3 0 3 1 3 2 3 ...
## $ Child           : int    1 2 1 1 0 1 2 0 2 3 ...
## $ Child_8         : int    1 0 0 0 0 0 0 0 0 1 ...
## $ Hobby           : int    0 0 1 0 0 1 1 1 0 1 ...
## $ Educ            : int    5 5 5 5 5 4 2 5 3 3 ...
## $ EducYrs         : num   19 15 16 17 17 13 11 17 13 13 ...
## $ ScaleS          : int    8 8 7 7 5 7 7 8 4 5 ...
## $ ScaleR          : int    6 5 3 5 4 7 7 4 4 5 ...
## $ ISL_cd          : int    4 3 2 3 2 2 2 3 3 3 ...
## $ ProfCateg       : int    2 2 NA NA 3 2 NA 3 4 NA ...
## $ DustCurrent     : int    1 NA 0 0 0 0 1 0 0 1 ...
## $ pDustAn         : int    NA 1 NA NA NA 2 NA NA NA NA ...
## $ Dust            : int    1 0 0 0 0 1 1 0 1 1 ...
## $ ChemCurrent     : int    1 NA 0 1 0 1 1 1 1 1 ...
## $ Chem            : int    1 1 0 1 0 1 1 1 1 1 ...
## $ BioCurrent      : int    0 NA 1 0 0 0 0 0 0 1 ...
## $ Bio             : int    0 0 1 0 0 0 0 0 0 1 ...
## $ ColdCurrent     : int    1 NA 1 1 1 1 1 1 1 1 ...
## $ Cold            : int    1 0 1 1 1 1 1 1 1 1 ...
## $ HeatCurrent     : int    1 NA 0 0 0 0 1 0 1 1 ...

```

```
## $ Heat : int 1 0 0 0 0 1 1 0 1 1 ...
## $ PhysLoadCurrent: int 1 NA 1 1 1 1 1 1 1 1 ...
## $ PhysLoad : int 1 0 1 1 1 1 1 1 1 1 ...
## $ DL : int 74 NA 60 70 70 66 46 NA 78 60 ...
## $ JD : int 21 NA 23 24 20 21 27 NA 23 22 ...
## $ CS : int 18 NA 17 20 17 18 10 NA 18 16 ...
## $ SS : int 13 NA NA 18 15 17 9 NA 18 15 ...
## $ SocSup : int 31 NA NA 38 32 35 19 NA 36 31 ...
## $ JobDis : num 0.733 NA 0.1 0 0.333 ...
## $ JI : int 14 NA 10 11 14 12 10 NA 12 11 ...
## $ JDJI : int 35 NA 33 35 34 33 37 NA 35 33 ...
## $ PE : int 10 NA 11 11 10 10 16 NA 13 12 ...
## $ JCQedcd : int 0 NA NA 0 1 0 1 NA 0 0 ...
## $ ERI_Ef_Ph : int 9 NA 10 8 13 7 20 NA 17 10 ...
## $ ERI_Ef_M : int 8 NA 9 7 12 6 16 NA 14 9 ...
## $ Overcom : int 11 NA 12 7 13 13 15 NA 18 12 ...
## $ SANx : int 33 46 42 38 49 29 53 NA 52 NA ...
## $ SANx_cd : int 2 3 2 2 3 1 3 NA 3 NA ...
## $ LE : int 108 53 71 109 181 133 101 NA 238 178 ...
## $ LE_cd : int 1 1 1 1 1 1 1 NA 2 1 ...
## $ FamilyPast : int 3 3 3 3 3 3 3 3 3 3 ...
## $ SMK : int 0 0 1 1 0 0 0 0 0 0 ...
## $ PasSMK : int 0 0 3 4 0 0 0 3 2 0 ...
## $ PARigMin : int 240 240 360 120 0 40 180 120 60 NA ...
## $ PAModMin : int 120 120 420 0 0 30 180 60 60 NA ...
## $ WalkMin : int 300 120 1800 240 0 180 120 120 60 120 ...
## $ FruVegs : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Omega3 : int 0 0 0 0 1 1 1 1 1 1 ...
## $ DPP4 : num NA NA 926 NA 1232 ...
## $ TGFb1 : num NA NA 76425 NA 71332 ...
## $ TSLP : num 0.485 NA 4.62 NA 4.62 NA NA 4.62 4.62 NA ...
## $ IL13 : num 414.7 NA 102.3 NA 28.1 ...
## $ IL17A : num NA NA 5.18 NA 5.18 NA NA 5.18 5.18 NA ...
## $ IL1b : num NA NA 2.06 NA 2.06 ...
## $ IL33 : num 14.55 NA 5.65 NA 1.79 ...
## $ IL4 : num 0.005 NA 1.31 NA 0.005 NA NA 0.005 0.005 NA ...
## $ IL5 : num 8.77 NA 22.51 NA 2.32 ...
## $ IL6 : num NA NA 46.05 NA 0.505 ...
## $ IL25 : num 0.08 NA 0.025 NA 0.025 NA NA 0.19 0.025 NA ...
## [list output truncated]
```

```
#summary(ba)
```

```
ba$BAcd <- replace(ba$BAcd,ba$BAcd == "", NA)
```

```
"Кодирование переменных как факторов"
```

```
## [1] "Кодирование переменных как факторов"
```

```
ba$Kazan <- as.factor(ba$Kazan)
```

```
ba$Point <- as.factor(ba$Point)
```

```
ba$BA <- as.factor(ba$BA)
```

```

#ba$set <- as.factor(ba$set)
ba$BAcd <- as.factor(ba$BAcd)
ba$BMId <- as.factor(ba$BMId)
ba$BAsevere <- as.factor(ba$BAsevere)
ba$BAobstr <- as.factor(ba$BAobstr)
ba$BAcontrol <- as.factor(ba$BAcontrol)
ba$BAdebut <- as.factor(ba$BAdebut)
ba$Atopia <- as.factor(ba$Atopia)
ba$Heredity <- as.factor(ba$Heredity)

ba$Sex <- as.factor(ba$Sex)
ba$Family <- as.factor(ba$Family)
ba$FamType <- as.factor(ba$FamType)
ba$Child <- as.factor(ba$Child)
ba$Child_8 <- as.factor(ba$Child_8)
ba$Hobby <- as.factor(ba$Hobby)
ba$Educ <- as.factor(ba$Educ)
ba$ScaleS <- as.factor(ba$ScaleS)
ba$ScaleR <- as.factor(ba$ScaleR)
ba$ISL_cd <- as.factor(ba$ISL_cd)

ba$ProfCateg <- as.factor(ba$ProfCateg)
ba$DustCurrent <- as.factor(ba$DustCurrent)
ba$Dust <- as.factor(ba$Dust)
ba$ChemCurrent <- as.factor(ba$ChemCurrent)
ba$Chem <- as.factor(ba$Chem)
ba$BioCurrent <- as.factor(ba$BioCurrent)
ba$Bio <- as.factor(ba$Bio)
ba$ColdCurrent <- as.factor(ba$ColdCurrent)
ba$Cold <- as.factor(ba$Cold)
ba$HeatCurrent <- as.factor(ba$HeatCurrent)
ba$Heat <- as.factor(ba$Heat)
ba$PhysLoadCurrent <- as.factor(ba$PhysLoadCurrent)
ba$PhysLoad <- as.factor(ba$PhysLoad)

ba$JCQedcd <- as.factor(ba$JCQedcd)
#ba$ERI_Phcd <- as.factor(ba$ERI_Phcd)
#ba$ERI_Mcd <- as.factor(ba$ERI_Mcd)
ba$SAnx_cd <- as.factor(ba$SAnx_cd)
ba$LE_cd <- as.factor(ba$LE_cd)
ba$FamilyPast <- as.factor(ba$FamilyPast)
#ba$Childhod <- as.factor(ba$Childhod)
#ba$Health_CD <- as.factor(ba$Health_CD)
#ba$Health_CD2 <- as.factor(ba$Health_CD2)
ba$SMK <- as.factor(ba$SMK)
ba$Food <- as.factor(ba$FruVegs)
ba$Omega3 <- as.factor(ba$Omega3)

table(ba$BAcd)

```

```
##
## j45.0 j45.1 j45.8      K
##    40    42    8    48

ba1 <- ba[ba$BAcd %in% c("j45.1", "K"), ]
str(ba1)

## 'data.frame':    90 obs. of  106 variables:
##  $ ID           : chr  "60" "74" "71" "58" ...
##  $ Kazan        : Factor w/ 2 levels "0","1": 1 2 1 2 1 2 2 1 2 2 ...
##  $ Point        : Factor w/ 10 levels "1","4","5","6",...: NA 1 NA 1 NA 1 2
NA 3 7 ...
##  $ TSPAvr       : num  NA 0.164 NA 0.164 NA 0.164 0.09 NA 0.107 0.166 ...
##  $ TSPMaxAvr    : num  NA 0.595 NA 0.595 NA ...
##  $ TSPMaxMax     : num  NA 0.734 NA 0.734 NA 0.734 0.437 NA 0.495 0.769 ...
##  $ PM10Avr      : num  NA 0.094 NA 0.094 NA 0.094 0.036 NA 0.043 0.089 ...
##  $ PM10MaxAvr   : num  NA 0.348 NA 0.348 NA ...
##  $ PM10MaxMax   : num  NA 0.44 NA 0.44 NA 0.44 0.285 NA 0.261 0.45 ...
##  $ PM2.5Avr     : num  NA 0.031 NA 0.031 NA 0.031 0.009 NA 0.01 0.029 ...
##  $ PM2.5MaxAvr  : num  NA 0.172 NA 0.172 NA ...
##  $ PM2.5MaxMax  : num  NA 0.253 NA 0.253 NA 0.253 0.157 NA 0.127 0.264 ...
##  $ TSPDT        : num  NA 0.038 NA 0.038 NA 0.038 0.026 NA 0.014 0.061 ...
##  $ PM10DT       : num  NA 0.03 NA 0.03 NA 0.03 0.02 NA 0.011 0.058 ...
##  $ PM25DT       : num  NA 0.028 NA 0.028 NA 0.028 0.018 NA 0.009 0.021 ...
##  $ Cu           : num  NA 0.56 NA 0.56 NA 0.56 NA NA 2.29 0.76 ...
##  $ C            : num  NA 93.4 NA 93.4 NA ...
##  $ LAL1         : num  NA 0.0277 NA 0.0277 NA 0.0277 NA NA 0.0139 0.0276
...
##  $ LAL2         : num  NA 0.0694 NA 0.0694 NA 0.0694 NA NA 0.0279 0.0553
...
##  $ TBPLogMax    : num  NA 0.0221 NA 0.0221 NA ...
##  $ PLogMax      : num  NA 0.0126 NA 0.0126 NA ...
##  $ TBLogMax     : num  NA 0.00948 NA 0.00948 NA ...
##  $ TBPLogAvr    : num  NA 0.00804 NA 0.00804 NA ...
##  $ PLogAvr      : num  NA 0.00457 NA 0.00457 NA ...
##  $ TBLogAvr     : num  NA 0.00346 NA 0.00346 NA ...
##  $ BA           : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 1 1 ...
##  $ BAcd         : Factor w/ 4 levels "j45.0","j45.1",...: 2 2 2 2 2 2 2 2 4
4 ...
##  $ BMI          : num  39 39.1 22.7 27.9 26.8 ...
##  $ BMIcd        : Factor w/ 3 levels "1","2","3": 3 3 1 2 2 3 2 1 1 2 ...
##  $ BAsevere     : Factor w/ 3 levels "1","2","3": 3 2 3 2 2 1 2 2 NA NA ...
##  $ BAobstr      : Factor w/ 2 levels "0","1": 1 2 2 1 1 1 1 1 NA NA ...
##  $ BAcontrol    : Factor w/ 4 levels "1","2","3","4": 2 2 NA 3 2 NA 1 2 NA
NA ...
##  $ BAdebut      : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 2 1 NA NA ...
##  $ IgE          : num  72.9 11.3 NA NA NA ...
##  $ Eos          : num  291.2 286 32.4 370 54 ...
##  $ Atopia       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Heredity     : Factor w/ 2 levels "0","1": 1 2 1 1 2 1 1 1 2 2 ...
##  $ Age          : num  56.7 37.7 58.7 54.7 39.9 ...
##  $ Sex          : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 1 2 2 1 ...
##  $ Family       : Factor w/ 6 levels "1","2","3","4",...: 6 6 6 2 6 1 5 6 3
```

```

6 ...
## $ FamType      : Factor w/ 5 levels "0","1","2","3",...: 4 4 4 3 4 4 4 4 3
4 ...
## $ Child        : Factor w/ 6 levels "0","1","2","3",...: 3 2 2 3 4 1 1 2 3
3 ...
## $ Child_8      : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 1 1 1 ...
## $ Hobby        : Factor w/ 2 levels "0","1": 1 2 2 1 2 1 1 1 2 2 ...
## $ Educ         : Factor w/ 5 levels "2","3","4","5",...: 4 4 3 2 2 4 4 4 1
3 ...
## $ EducYrs      : num  15 16 13 13 13 16 15 17 10 13 ...
## $ ScaleS       : Factor w/ 9 levels "1","3","4","5",...: 7 6 6 3 4 3 5 2 4
6 ...
## $ ScaleR       : Factor w/ 10 levels "1","2","3","4",...: 5 3 7 4 5 5 4 3 5
7 ...
## $ ISL_cd       : Factor w/ 4 levels "1","2","3","4": 3 2 2 3 3 3 2 2 3 2
...
## $ ProfCateg    : Factor w/ 5 levels "1","2","3","4",...: 2 NA 2 4 NA 2 5 NA
4 4 ...
## $ DustCurrent  : Factor w/ 2 levels "0","1": NA 1 1 1 2 1 NA 1 2 2 ...
## $ pDustAn      : int   1 NA 2 NA NA 0 5 5 3 3 ...
## $ Dust         : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 2 2 2 2 ...
## $ ChemCurrent  : Factor w/ 2 levels "0","1": NA 1 2 2 2 1 NA 2 2 1 ...
## $ Chem         : Factor w/ 2 levels "0","1": 2 1 2 2 2 1 2 2 2 2 ...
## $ BioCurrent   : Factor w/ 2 levels "0","1": NA 2 1 1 2 1 NA 1 1 1 ...
## $ Bio          : Factor w/ 2 levels "0","1": 1 2 1 1 2 1 2 1 1 1 ...
## $ ColdCurrent  : Factor w/ 2 levels "0","1": NA 2 2 2 2 1 NA 2 2 2 ...
## $ Cold         : Factor w/ 2 levels "0","1": 1 2 2 2 2 1 2 2 2 2 ...
## $ HeatCurrent  : Factor w/ 2 levels "0","1": NA 1 1 2 2 1 NA 2 2 1 ...
## $ Heat         : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 1 2 2 1 ...
## $ PhysLoadCurrent: Factor w/ 2 levels "0","1": NA 2 2 2 2 1 NA 2 2 1 ...
## $ PhysLoad     : Factor w/ 2 levels "0","1": 1 2 2 2 2 1 2 2 2 1 ...
## $ DL           : int   NA 60 66 78 60 76 NA 58 NA 64 ...
## $ JD           : int   NA 23 21 23 22 23 NA 32 NA 21 ...
## $ CS           : int   NA 17 18 18 16 12 NA 17 NA 16 ...
## $ SS           : int   NA NA 17 18 15 14 NA 7 NA 16 ...
## $ SocSup       : int   NA NA 35 36 31 26 NA 24 NA 32 ...
## $ JobDis       : num   NA 0.1 0 0.1 0 ...
## $ JI           : int   NA 10 12 12 11 16 NA 15 NA NA ...
## $ JDJI        : int   NA 33 33 35 33 39 NA 47 NA NA ...
## $ PE           : int   NA 11 10 13 12 8 NA 9 NA 11 ...
## $ JCQedcd      : Factor w/ 2 levels "0","1": NA NA 1 1 1 1 NA 1 NA 1 ...
## $ ERI_Ef_Ph    : int   NA 10 7 17 10 17 NA 20 7 7 ...
## $ ERI_Ef_M     : int   NA 9 6 14 9 16 NA 19 5 6 ...
## $ Overcom      : int   NA 12 13 18 12 16 NA 18 11 14 ...
## $ SANx         : int   46 42 29 52 NA 42 52 NA 44 45 ...
## $ SANx_cd      : Factor w/ 3 levels "1","2","3": 3 2 1 3 NA 2 3 NA 2 3 ...
## $ LE           : int   53 71 133 238 178 265 71 237 100 0 ...
## $ LE_cd        : Factor w/ 3 levels "1","2","3": 1 1 1 2 1 2 1 2 1 1 ...
## $ FamilyPast   : Factor w/ 2 levels "2","3": 2 2 2 2 2 2 2 2 2 2 ...
## $ SMK          : Factor w/ 3 levels "0","1","2": 1 2 1 1 1 1 1 2 1 1 ...
## $ PasSMK       : int   0 3 0 2 0 0 0 2 0 0 ...
## $ PARigMin     : int   240 360 40 60 NA NA 300 0 NA NA ...

```

```
## $ PAModMin      : int  120 420 30 60 NA NA 120 0 NA NA ...
## $ WalkMin       : int  120 1800 180 60 120 NA 240 50 NA NA ...
## $ FruVegs       : int   0 0 0 0 0 1 0 0 1 1 ...
## $ Omega3        : Factor w/ 2 levels "0","1": 1 1 2 2 2 2 1 1 2 2 ...
## $ DPP4          : num  NA 926 NA 1069 NA ...
## $ TGFb1         : num  NA 76425 NA 99988 NA ...
## $ TSLP          : num  NA 4.62 NA 4.62 NA ...
## $ IL13          : num  NA 102.3 NA 40.7 NA ...
## $ IL17A         : num  NA 5.18 NA 5.18 NA 5.18 5.18 NA NA 5.18 ...
## $ IL1b          : num  NA 2.06 NA 2.06 NA ...
## $ IL33          : num  NA 5.65 NA 1.79 NA ...
## $ IL4           : num  NA 1.31 NA 0.005 NA 6.21 0.005 NA NA 0.03 ...
## $ IL5           : num  NA 22.51 NA 2.32 NA ...
## $ IL6           : num  NA 46 NA 1.4 NA ...
## $ IL25          : num  NA 0.025 NA 0.025 NA 0.025 0.025 NA NA 0.025 ...
## [list output truncated]
```

```
table(ba1$BAcd)
```

```
##
## j45.0 j45.1 j45.8      K
##      0    42      0    48
```

```
ba2 <- ba[ba$BAcd %in% c("j45.0", "K"), ]
str(ba2)
```

```
## 'data.frame':    88 obs. of  106 variables:
## $ ID             : chr  "3" "52" "81" "86" ...
## $ Kazan          : Factor w/ 2 levels "0","1": 1 1 1 2 1 2 2 2 2 2 ...
## $ Point          : Factor w/ 10 levels "1","4","5","6",...: NA NA NA 10 NA 1
##                  1 3 7 3 ...
## $ TSPAavr        : num  NA NA NA 0.15 NA 0.164 0.164 0.107 0.166 0.107 ...
## $ TSPMaxAvr      : num  NA NA NA 0.552 NA ...
## $ TSPMaxMax      : num  NA NA NA 0.72 NA 0.734 0.734 0.495 0.769 0.495 ...
## $ PM10Avr        : num  NA NA NA 0.081 NA 0.094 0.094 0.043 0.089 0.043 ...
## $ PM10MaxAvr     : num  NA NA NA 0.277 NA ...
## $ PM10MaxMax     : num  NA NA NA 0.42 NA 0.44 0.44 0.261 0.45 0.261 ...
## $ PM2.5Avr       : num  NA NA NA 0.025 NA 0.031 0.031 0.01 0.029 0.01 ...
## $ PM2.5MaxAvr    : num  NA NA NA 0.142 NA ...
## $ PM2.5MaxMax    : num  NA NA NA 0.379 NA 0.253 0.253 0.127 0.264 0.127 ...
## $ TSPDT          : num  NA NA NA 0.058 NA 0.038 0.038 0.014 0.061 0.014 ...
## $ PM10DT         : num  NA NA NA 0.046 NA 0.03 0.03 0.011 0.058 0.011 ...
## $ PM25DT         : num  NA NA NA 0.026 NA 0.028 0.028 0.009 0.021 0.009 ...
## $ Cu             : num  NA NA NA NA NA 0.56 0.56 2.29 0.76 2.29 ...
## $ C              : num  NA NA NA 77.6 NA ...
## $ LAL1           : num  NA NA NA 0.0064 NA 0.0277 0.0277 0.0139 0.0276
##                  0.0139 ...
## $ LAL2           : num  NA NA NA 0.0382 NA 0.0694 0.0694 0.0279 0.0553
##                  0.0279 ...
## $ TBPLogMax      : num  NA NA NA 0.0188 NA ...
## $ PLogMax        : num  NA NA NA 0.0105 NA ...
## $ TBLogMax       : num  NA NA NA 0.00829 NA ...
## $ TBPLogAvr      : num  NA NA NA 0.00759 NA ...
```

```

## $ PLogAvr      : num  NA NA NA 0.00429 NA ...
## $ TBlogAvr    : num  NA NA NA 0.0033 NA ...
## $ BA          : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 1 1 1 ...
## $ BAcid       : Factor w/ 4 levels "j45.0","j45.1",...: 1 1 1 1 1 1 1 4 4
4 ...
## $ BMI         : num  27.5 25.9 29.4 23.4 NA ...
## $ BMIdc       : Factor w/ 3 levels "1","2","3": 2 2 2 1 NA 1 1 1 2 2 ...
## $ BAsevere    : Factor w/ 3 levels "1","2","3": 2 2 3 1 3 2 3 NA NA NA
...
## $ BAobstr     : Factor w/ 2 levels "0","1": 2 1 2 1 2 2 2 NA NA NA ...
## $ BAcontrol   : Factor w/ 4 levels "1","2","3","4": 2 NA 3 1 3 NA NA NA
NA NA ...
## $ BAdebut     : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 1 NA NA NA ...
## $ IgE         : num  155 NA 51.1 555 NA ...
## $ Eos         : num  276 510.3 270 423 97.5 ...
## $ Atopia      : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 1 1 1 ...
## $ Heredity    : Factor w/ 2 levels "0","1": 2 1 1 1 1 2 1 2 2 1 ...
## $ Age         : num  39.7 38.8 46.6 26.4 41.8 ...
## $ Sex         : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 2 1 2 ...
## $ Family      : Factor w/ 6 levels "1","2","3","4",...: 6 6 6 1 5 1 6 3 6
6 ...
## $ FamType     : Factor w/ 5 levels "0","1","2","3",...: 4 4 2 4 2 4 4 3 4
4 ...
## $ Child       : Factor w/ 6 levels "0","1","2","3",...: 2 2 3 1 3 1 2 3 3
3 ...
## $ Child_8     : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 1 1 1 1 ...
## $ Hobby       : Factor w/ 2 levels "0","1": 1 1 2 2 1 1 1 2 2 1 ...
## $ Educ        : Factor w/ 5 levels "2","3","4","5",...: 4 4 1 4 2 1 4 1 3
3 ...
## $ EducYrs     : num  19 17 11 17 13 11.5 18 10 13 12 ...
## $ ScaleS      : Factor w/ 9 levels "1","3","4","5",...: 7 6 6 7 4 4 6 4 6
4 ...
## $ ScaleR      : Factor w/ 10 levels "1","2","3","4",...: 6 5 7 4 5 3 4 5 7
5 ...
## $ ISL_cd      : Factor w/ 4 levels "1","2","3","4": 4 3 2 3 2 2 3 3 2 3
...
## $ ProfCateg   : Factor w/ 5 levels "1","2","3","4",...: 2 NA NA 3 NA 3 3 4
4 4 ...
## $ DustCurrent : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 2 2 1 ...
## $ pDustAn     : int  NA NA NA NA NA NA 0 3 3 5 ...
## $ Dust        : Factor w/ 2 levels "0","1": 2 1 2 1 2 1 2 2 2 2 ...
## $ ChemCurrent : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 1 2 1 2 ...
## $ Chem        : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 1 2 2 2 ...
## $ BioCurrent  : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ Bio         : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ ColdCurrent : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 2 2 ...
## $ Cold        : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 2 2 ...
## $ HeatCurrent : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 2 1 2 ...
## $ Heat        : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 2 1 2 ...
## $ PhysLoadCurrent: Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 1 2 ...
## $ PhysLoad    : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 1 2 ...
## $ DL          : int  74 70 46 NA 68 70 52 NA 64 64 ...

```



```
## $ JD : int 21 24 27 NA 26 17 29 NA 21 24 ...
## $ CS : int 18 20 10 NA 21 17 14 NA 16 16 ...
## $ SS : int 13 18 9 NA 19 18 14 NA 16 14 ...
## $ SocSup : int 31 38 19 NA 40 35 28 NA 32 30 ...
## $ JobDis : num 0.733 0 0.2 NA 0.533 ...
## $ JI : int 14 11 10 NA 14 17 11 NA NA 12 ...
## $ JDJI : int 35 35 37 NA 40 34 40 NA NA 36 ...
## $ PE : int 10 11 16 NA 10 7 11 NA 11 14 ...
## $ JCQedcd : Factor w/ 2 levels "0","1": 1 1 2 NA 1 1 1 NA 1 1 ...
## $ ERI_Ef_Ph : int 9 8 20 NA 9 6 6 7 7 7 ...
## $ ERI_Ef_M : int 8 7 16 NA 8 5 5 5 6 6 ...
## $ Overcom : int 11 7 15 NA 11 7 13 11 14 9 ...
## $ SANx : int 33 38 53 NA 52 69 56 44 45 38 ...
## $ SANx_cd : Factor w/ 3 levels "1","2","3": 2 2 3 NA 3 3 3 2 3 2 ...
## $ LE : int 108 109 101 NA 53 120 217 100 0 23 ...
## $ LE_cd : Factor w/ 3 levels "1","2","3": 1 1 1 NA 1 1 2 1 1 1 ...
## $ FamilyPast : Factor w/ 2 levels "2","3": 2 2 2 2 2 2 2 2 2 ...
## $ SMK : Factor w/ 3 levels "0","1","2": 1 2 1 1 1 1 3 1 1 1 ...
## $ PasSMK : int 0 4 0 3 0 0 2 0 0 0 ...
## $ PARigMin : int 240 120 180 120 0 NA NA NA NA 60 ...
## $ PAModMin : int 120 0 180 60 120 NA NA NA NA 120 ...
## $ WalkMin : int 300 240 120 120 180 NA NA NA NA 120 ...
## $ FruVegs : int 0 0 0 0 0 1 1 1 1 0 ...
## $ Omega3 : Factor w/ 2 levels "0","1": 1 1 2 2 2 2 2 2 2 1 ...
## $ DPP4 : num NA NA NA 1116 NA ...
## $ TGFb1 : num NA NA NA 57225 NA ...
## $ TSLP : num 0.485 NA NA 4.62 NA 4.62 4.62 NA 4.62 NA ...
## $ IL13 : num 414.7 NA NA 69.1 NA ...
## $ IL17A : num NA NA NA 5.18 NA 5.18 5.18 NA 5.18 NA ...
## $ IL1b : num NA NA NA 2.06 NA ...
## $ IL33 : num 14.6 NA NA 200.2 NA ...
## $ IL4 : num 0.005 NA NA 0.005 NA 0.005 0.08 NA 0.03 NA ...
## $ IL5 : num 8.77 NA NA 2.32 NA ...
## $ IL6 : num NA NA NA 0.505 NA 0.505 4.65 NA 2.92 NA ...
## $ IL25 : num 0.08 NA NA 0.19 NA 0.025 0.025 NA 0.025 NA ...
## [list output truncated]
```

```
table(ba2$BAcd)
```

```
##
## j45.0 j45.1 j45.8 K
## 40 0 0 48
```

#Создание листов с переменными

##лист exp_n содержит численные экспозиционные переменные и кофандеры"

##названия переменных экспозиции

```
expnames<-c("TSPAvr", "TSPMaxAvr", "TSPMaxMax", "PM10Avr", "PM10MaxAvr",
            "PM10MaxMax", "PM2.5Avr", "PM2.5MaxAvr", "PM2.5MaxMax", "TSPDT",
            "PM10DT", "PM25DT", "TBPLogMax", "PLogMax", "TBLogMax",
            "TBPLogAvr", "PLogAvr", "TBLogAvr",
            "BMI", "Age", "EducYrs",
```

```

      "DL", "JD", "CS", "SS", "SocSup", "JobDis", "JI", "PE",
      "ERI_Ef_Ph", "ERI_Ef_M",
      "Overcom", "SAnx",
      "PasSMK", "PA_RigMin", "PA_ModMin", "Walk_Min")

exp_n1 <-
list(ba1$TSPAvr, ba1$TSPMaxAvr, ba1$TSPMaxMax, ba1$PM10Avr, ba1$PM10MaxAvr,

ba1$PM10MaxMax, ba1$PM2.5Avr, ba1$PM2.5MaxAvr, ba1$PM2.5MaxMax, ba1$TSPDT,
      ba1$PM10DT, ba1$PM25DT, ba1$TBPLogMax, ba1$PLogMax, ba1$TBLogMax,
      ba1$TBPLogAvr, ba1$PLogAvr, ba1$TBLogAvr,
      ba1$BMI, ba1$Age, ba1$EducYrs,
      ba1$DL, ba1$JD, ba1$CS, ba1$SS, ba1$SocSup, ba1$JobDis, ba1$JI, ba1$PE,
      ba1$ERI_Ef_Ph, ba1$ERI_Ef_M,
      ba1$Overcom, ba1$SAnx, ba1$PasSMK, ba1$PARigMin, ba1$PAModMin,
      ba1$WalkMin)

exp_n2 <-
list(ba2$TSPAvr, ba2$TSPMaxAvr, ba2$TSPMaxMax, ba2$PM10Avr, ba2$PM10MaxAvr,

ba2$PM10MaxMax, ba2$PM2.5Avr, ba2$PM2.5MaxAvr, ba2$PM2.5MaxMax, ba2$TSPDT,
      ba2$PM10DT, ba2$PM25DT, ba2$TBPLogMax, ba2$PLogMax, ba2$TBLogMax,
      ba2$TBPLogAvr, ba2$PLogAvr, ba2$TBLogAvr,
      ba2$BMI, ba2$Age, ba2$EducYrs,
      ba2$DL, ba2$JD, ba2$CS, ba2$SS, ba2$SocSup, ba2$JobDis, ba2$JI, ba2$PE,
      ba2$ERI_Ef_Ph, ba2$ERI_Ef_M,
      ba2$Overcom, ba2$SAnx, ba2$PasSMK, ba2$PARigMin, ba2$PAModMin,
      ba2$WalkMin)

names(exp_n1)<-expnames
names(exp_n2)<-expnames

#Сравнение больных и контроля по количественным переменным

#функция для проверки на нормальность и т-теста
# rquery.t.test
# Description : Performs one or two samples t-test
# x : a (non-empty) numeric vector of data values.
# y : an optional (non-empty) numeric vector of data values
# paired : if TRUE, paired t-test is performed
# graph : if TRUE, the distribution of the data is shown
# for the inspection of normality
# ... : further arguments to be passed to the built-in t.test() R function
# 1. shapiro.test is used to check normality
# 2. F-test is performed to check equality of variances
# If the variances are different, then Welch t-test is used

rquery.t.test<-function(x, y = NULL, paired = FALSE, graph = TRUE, ...)
{
  # I. Preliminary test : normality and variance tests
  # ~~~~~

```

```

var.equal = FALSE # by default

# I.1 One sample t test
if(is.null(y)){
  if(graph) par(mfrow=c(1,2))
  shapiro.px<-normaTest(x, graph,
                        hist.title="X - Histogram",
                        qq.title="X - Normal Q-Q Plot")
  if(shapiro.px < 0.05)
    warning("x is not normally distributed :",
            " Shapiro-Wilk test p-value : ", shapiro.px,
            ".\n Use a non-parametric test like Wilcoxon test.")
}

# I.2 Two samples t test
if(!is.null(y)){

  # I.2.a unpaired t test
  if(!paired){
    if(graph) par(mfrow=c(2,2))
    # normality test
    shapiro.px<-normaTest(x, graph,
                          hist.title="X - Histogram",
                          qq.title="X - Normal Q-Q Plot")
    shapiro.py<-normaTest(y, graph,
                          hist.title="Y - Histogram",
                          qq.title="Y - Normal Q-Q Plot")
    if(shapiro.px < 0.05 | shapiro.py < 0.05){
      warning("x or y is not normally distributed :",
              " Shapiro test p-value : ", shapiro.px,
              " (for x) and ", shapiro.py, " (for y)",
              ".\n Use a non parametric test like Wilcoxon test.")
    }
    # Check for equality of variances
    if(var.test(x,y)$p.value >= 0.05) var.equal=TRUE
  }

  # I.2.b Paired t-test
  else {
    if(graph) par(mfrow=c(1,2))
    d = x-y
    shapiro.pd<-normaTest(d, graph,
                          hist.title="D - Histogram",
                          qq.title="D - Normal Q-Q Plot")
    if(shapiro.pd < 0.05 )
      warning("The difference d ( = x-y) is not normally distributed :",
              " Shapiro-Wilk test p-value : ", shapiro.pd,
              ".\n Use a non-parametric test like Wilcoxon test.")
  }
}
}

```

```

# II. Student's t-test
# ~~~~~
res <- t.test(x, y, paired=paired, var.equal=var.equal, ...)
return(res)
}

# Helper function
# Performs normality test using Shapiro Wilk's method
# The histogram and Q-Q plot of the data are plotted
# x : a (non-empty) numeric vector of data values.
# graph : possible values are TRUE or FALSE. If TRUE,
# the histogram and the Q-Q plot of the data are displayed
# hist.title : title of the histogram
# qq.title : title of the Q-Q plot
normaTest<-function(x, graph=TRUE,
                    hist.title="Histogram",
                    qq.title="Normal Q-Q Plot",...)
{
  # Significance test
  shapiro.p<-signif(shapiro.test(x)$p.value,1)

  if(graph){
    # Plot : Visual inspection
    h<-hist(x, col="lightblue", main=hist.title,
           xlab="Data values", ...)
    m<-round(mean(x),1)
    s<-round(sd(x),1)
    mtext(paste0("Mean : ", m, "; SD : ", s),
          side=3, cex=0.8)
    # add normal curve
    xfit<-seq(min(x),max(x),length=40)
    yfit<-dnorm(xfit,mean=mean(x),sd=sd(x))
    yfit <- yfit*diff(h$mids[1:2])*length(x)
    lines(xfit, yfit, col="red", lwd=2)
    # qq plot
    qqnorm(x, pch=19, frame.plot=FALSE,main=qq.title)
    qqline(x)
    mtext(paste0("Shapiro-Wilk, p-val : ", shapiro.p),
          side=3, cex=0.8)
  }
  return(shapiro.p)
}

#получение описательных статистик количественных переменных
f.numstat <- function (x) {
  summary(x)
}

f.sd <- function (x) {
  sd(x, na.rm=T)
}

```

```
lapply(exp_n1, f.numstat)
```

```
## $TSPAavr
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.0900 0.1140 0.1500 0.1419 0.1640 0.1660     29
##
## $TSPMaxAavr
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.2750 0.4998 0.5518 0.5305 0.5952 0.6332     29
##
## $TSPMaxMax
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.4370 0.6720 0.7340 0.6897 0.7480 0.7690     29
##
## $PM10Aavr
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.03600 0.04800 0.08100 0.07357 0.08900 0.09400     29
##
## $PM10MaxAavr
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.1503 0.2025 0.2773 0.2838 0.3485 0.3797     29
##
## $PM10MaxMax
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.261 0.291 0.430 0.393 0.440 0.453     29
##
## $PM2.5Aavr
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.00900 0.01200 0.02500 0.02298 0.03100 0.03200     29
##
## $PM2.5MaxAavr
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.0583 0.1062 0.1555 0.1413 0.1720 0.1985     29
##
## $PM2.5MaxMax
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.1270 0.1570 0.2400 0.2328 0.2530 0.3790     29
##
## $TSPDT
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.01400 0.01600 0.03800 0.03703 0.05800 0.06100     29
##
## $PM10DT
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.01000 0.01100 0.03000 0.02975 0.04600 0.05800     29
##
## $PM25DT
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
## 0.0060 0.0090 0.0210 0.0192 0.0260 0.0280     29
##
## $TBPLogMax
```

```

##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 0.008971 0.013276 0.018778 0.017698 0.022095 0.022998        29
##
## $PLogMax
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 0.004758 0.007416 0.010625 0.009962 0.012611 0.012901        29
##
## $TBlogMax
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 0.004213 0.005860 0.008291 0.007736 0.009484 0.010097        29
##
## $TBPLogAvr
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 0.002101 0.003168 0.007587 0.006117 0.007962 0.008038        30
##
## $PLogAvr
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 0.001510 0.001812 0.004286 0.003478 0.004496 0.004569        30
##
## $TBlogAvr
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 0.001012 0.001615 0.003300 0.002717 0.003457 0.003462        30
##
## $BMI
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 17.30    21.93    25.24    26.04    28.40    46.43         1
##
## $Age
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.
## 17.37    26.23    40.08    41.50    55.76    65.91
##
## $EducYrs
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 6.00     13.00    15.00    15.08    17.00    24.00        25
##
## $DL
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 40.00    58.00    61.00    64.25    66.00    94.00        50
##
## $JD
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 14.00    21.75    23.00    22.98    25.00    32.00        50
##
## $CS
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 12.00    15.00    17.00    16.75    18.00    24.00        50
##
## $SS
##      Min.  1st Qu.  Median      Mean  3rd Qu.      Max.      NA's
## 7.00     13.25    15.00    14.68    16.00    25.00        52
##
## $SocSup

```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      23.00  29.00   31.00   31.57  34.00   43.00     53
##
## $JobDis
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      0.0000  0.1000  0.3333  0.3569  0.6333  0.9333     49
##
## $JI
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      10.00  12.00   12.00   13.05  14.00   18.00     53
##
## $PE
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      5.00   9.75   11.00   11.12  13.00   18.00     50
##
## $ERI_Ef_Ph
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      6.00   7.00   10.00   10.62  12.00   21.00     53
##
## $ERI_Ef_M
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      5.000  6.000   8.000   8.919 10.000  19.000     53
##
## $Overcom
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      9.00  12.00   13.00   13.74  16.00   18.00     52
##
## $SAnx
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      20.00  41.00   45.00   44.63  51.00   66.00     31
##
## $PasSMK
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      0.0     0.0     0.0     0.3     0.0     3.0     60
##
## $PA_RigMin
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      0.00   0.00   40.00   80.86 120.00  360.00     69
##
## $PA_ModMin
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      0.00   0.00   60.00   89.76 120.00  420.00     69
##
## $Walk_Min
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      0.0    52.5   120.0   193.6  165.0  1800.0     68

lapply(exp_n2, f.numstat)

## $TSPAavr
##      Min. 1st Qu.  Median      Mean 3rd Qu.  Max.    NA's
##      0.1070  0.1150  0.1500  0.1433  0.1640  0.1660     22
##
```

```

## $TSPMaxAvr
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.3623 0.4998 0.5518 0.5351 0.5952 0.6332      22
##
## $TSPMaxMax
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.4950 0.6800 0.7340 0.6975 0.7480 0.7690      22
##
## $PM10Avr
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.04300 0.05500 0.08100 0.07495 0.09275 0.09400      22
##
## $PM10MaxAvr
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.1715 0.2167 0.2773 0.2871 0.3485 0.3797      22
##
## $PM10MaxMax
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.2610 0.4150 0.4300 0.3989 0.4400 0.4530      22
##
## $PM2.5Avr
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.01000 0.01400 0.02500 0.02362 0.03100 0.03200      22
##
## $PM2.5MaxAvr
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.0583 0.1170 0.1555 0.1442 0.1720 0.1985      22
##
## $PM2.5MaxMax
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.127 0.221 0.240 0.234 0.253 0.379      22
##
## $TSPDT
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.01400 0.01450 0.03800 0.03521 0.05800 0.06100      22
##
## $PM10DT
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.01000 0.01100 0.03000 0.02814 0.04600 0.05800      22
##
## $PM25DT
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.00600 0.00900 0.02100 0.01859 0.02750 0.02800      22
##
## $TBPLogMax
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.008971 0.015642 0.018778 0.017952 0.022095 0.022998      22
##
## $PLogMax
##   Min. 1st Qu.  Median      Mean 3rd Qu.    Max.    NA's
## 0.004758 0.008914 0.010625 0.010130 0.012611 0.012901      22
##

```



```

## $TBlogMax
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.004213 0.006728 0.008291 0.007822 0.009484 0.010097      22
##
## $TBPLogAvr
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.002546 0.004117 0.007587 0.006239 0.008038 0.008038      23
##
## $PLogAvr
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.001545 0.002272 0.004286 0.003531 0.004569 0.004569      23
##
## $TBlogAvr
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.001243 0.001845 0.003300 0.002773 0.003462 0.003462      23
##
## $BMI
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 16.41    20.78    24.14    24.35    27.15    36.74         2
##
## $Age
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
## 17.37    23.25    33.08    36.02    45.81    64.89
##
## $EducYrs
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 6.00     13.00    16.00    15.18    18.00    24.00      23
##
## $DL
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 40.0     58.0     62.0     63.5     70.0     94.0       48
##
## $JD
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 14.00    21.00    23.00    23.18    25.25    31.00      48
##
## $CS
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 10.00    15.00    16.50    16.62    18.00    24.00      48
##
## $SS
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 7.00     14.00    15.00    14.85    17.00    20.00      49
##
## $SocSup
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 18.00    29.25    32.00    31.61    34.75    41.00      50
##
## $JobDis
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.0000    0.2000    0.3000    0.3398    0.5333    0.9000      47
##

```

```
## $JI
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   10.00  12.00   13.00   13.08  13.00   18.00     51
##
## $PE
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   5.00   9.75   11.00   11.12  14.00   18.00     48
##
## $ERI_Ef_Ph
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   6.000   7.000   8.000   9.897  11.500  21.000     49
##
## $ERI_Ef_M
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   5.000   5.000   7.000   8.282  10.000  16.000     49
##
## $Overcom
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   7.00   11.75   13.00   13.10  15.00   18.00     48
##
## $SAnx
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   20.00  38.50   45.00   44.36  51.50   69.00     29
##
## $PasSMK
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   0.0000  0.0000  0.0000  0.5862  0.0000  4.0000     59
##
## $PA_RigMin
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   0.00   0.00   60.00   83.39  120.00  240.00     65
##
## $PA_ModMin
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   0.0    0.0    80.0   100.7  180.0   420.0     67
##
## $Walk_Min
##   Min. 1st Qu.  Median    Mean 3rd Qu.  Max.    NA's
##   0.0    55.0   120.0   135.2  180.0   480.0     64
```

*#подготовительная работа -
#разделение на группы по группирующей переменной
#и создание листа spl (листы, каждый из которых состоит из 2-х листов с именами)
#раздельно для j45.0 и j45.1*

```
num1<-length(exp_n1)
spl.exp_n1<-vector(num1,mode="list")
names(spl.exp_n1)<-c("TSPAвр", "TSPMaxAвр", "TSPMaxMax", "PM10Aвр", "PM10MaxAвр",
                    "PM10MaxMax", "PM2.5Aвр", "PM2.5MaxAвр", "PM2.5MaxMax", "TSPDT",
                    "PM10DT", "PM25DT", "TBPLogMax", "PLogMax", "TBLogMax",
                    "TBPLogAвр", "PLogAвр", "TBLogAвр",
                    "BMI", "Age", "EducYrs",
```

```

        "DL", "JD", "CS", "SS", "SocSup", "JobDis", "JI", "PE",
        "ERI_Ef_Ph", "ERI_Ef_M",
        "Overcom", "SAnx",
        "PasSMK", "PA_RigMin", "PA_ModMin", "Walk_Min")
for (i in 1:num1) {
  spl.exp_n1[[i]]<-split(exp_n1[[i]],ba1$BA)
}

num2<-length(exp_n2)
spl.exp_n2<-vector(num2,mode="list")
names(spl.exp_n2)<-c("TSPAvr", "TSPMaxAvr", "TSPMaxMax", "PM10Avr", "PM10MaxAvr",
  "PM10MaxMax", "PM2.5Avr", "PM2.5MaxAvr", "PM2.5MaxMax", "TSPDT",
  "PM10DT", "PM25DT", "TBPLogMax", "PLogMax", "TBLogMax",
  "TBPLogAvr", "PLogAvr", "TBLogAvr",
  "BMI", "Age", "EducYrs",
  "DL", "JD", "CS", "SS", "SocSup", "JobDis", "JI", "PE",
  "ERI_Ef_Ph", "ERI_Ef_M",
  "Overcom", "SAnx",
  "PasSMK", "PA_RigMin", "PA_ModMin", "Walk_Min")
for (i in 1:num2) {
  spl.exp_n2[[i]]<-split(exp_n2[[i]],ba2$BA)
}

##листы, которые содержат переменные исхода только для одной группы
exp_n.BA11<-lapply(spl.exp_n1, "[[", "1")
names(exp_n.BA11)<-c("TSPAvr", "TSPMaxAvr", "TSPMaxMax", "PM10Avr", "PM10MaxAvr",
  "PM10MaxMax", "PM2.5Avr", "PM2.5MaxAvr", "PM2.5MaxMax", "TSPDT",
  "PM10DT", "PM25DT", "TBPLogMax", "PLogMax", "TBLogMax",
  "TBPLogAvr", "PLogAvr", "TBLogAvr",
  "BMI", "Age", "EducYrs",
  "DL", "JD", "CS", "SS", "SocSup", "JobDis", "JI", "PE",
  "ERI_Ef_Ph", "ERI_Ef_M",
  "Overcom", "SAnx",
  "PasSMK", "PA_RigMin", "PA_ModMin", "Walk_Min")

##листы, которые содержат переменные исхода только для другой группы
exp_n.BA01<-lapply(spl.exp_n1, "[[", "0")
names(exp_n.BA01)<-c("TSPAvr", "TSPMaxAvr", "TSPMaxMax", "PM10Avr", "PM10MaxAvr",
  "PM10MaxMax", "PM2.5Avr", "PM2.5MaxAvr", "PM2.5MaxMax", "TSPDT",
  "PM10DT", "PM25DT", "TBPLogMax", "PLogMax", "TBLogMax",
  "TBPLogAvr", "PLogAvr", "TBLogAvr",
  "BMI", "Age", "EducYrs",
  "DL", "JD", "CS", "SS", "SocSup", "JobDis", "JI", "PE",
  "ERI_Ef_Ph", "ERI_Ef_M",
  "Overcom", "SAnx",
  "PasSMK", "PA_RigMin", "PA_ModMin", "Walk_Min")

exp_n.BA11<-lapply(exp_n.BA11,function(x) x[!is.na(x)])
exp_n.BA01<-lapply(exp_n.BA01,function(x) x[!is.na(x)])

exp_n.BA12<-lapply(spl.exp_n2, "[[", "1")
names(exp_n.BA12)<-c("TSPAvr", "TSPMaxAvr", "TSPMaxMax", "PM10Avr", "PM10MaxAvr",

```

```

"PM10MaxMax", "PM2.5Avr", "PM2.5MaxAvr", "PM2.5MaxMax", "TSPDT",
"PM10DT", "PM25DT", "TBPLogMax", "PLogMax", "TBLogMax",
"TBPLogAvr", "PLogAvr", "TBLogAvr",
"BMI", "Age", "EducYrs",
"DL", "JD", "CS", "SS", "SocSup", "JobDis", "JI", "PE",
"ERI_Ef_Ph", "ERI_Ef_M",
"Overcom", "SAnx",
"PasSMK", "PA_RigMin", "PA_ModMin", "Walk_Min")

```

##листы, которые содержат переменные исхода только для другой группы

```

exp_n.BA02<-lapply(spl.exp_n2, "[", "0")
names(exp_n.BA02)<-c("TSPAvr", "TSPMaxAvr", "TSPMaxMax", "PM10Avr", "PM10MaxAvr",
"PM10MaxMax", "PM2.5Avr", "PM2.5MaxAvr", "PM2.5MaxMax", "TSPDT",
"PM10DT", "PM25DT", "TBPLogMax", "PLogMax", "TBLogMax",
"TBPLogAvr", "PLogAvr", "TBLogAvr",
"BMI", "Age", "EducYrs",
"DL", "JD", "CS", "SS", "SocSup", "JobDis", "JI", "PE",
"ERI_Ef_Ph", "ERI_Ef_M",
"Overcom", "SAnx",
"PasSMK", "PA_RigMin", "PA_ModMin", "Walk_Min")

```

```

exp_n.BA12<-lapply(exp_n.BA12, function(x) x[!is.na(x)])
exp_n.BA02<-lapply(exp_n.BA02, function(x) x[!is.na(x)])

```

#T-mecm

```

for (i in 1:num1) {
  print(names(exp_n1[i]))
  print(rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]), alternative = "g")
}

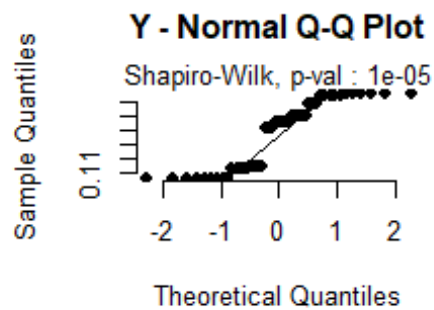
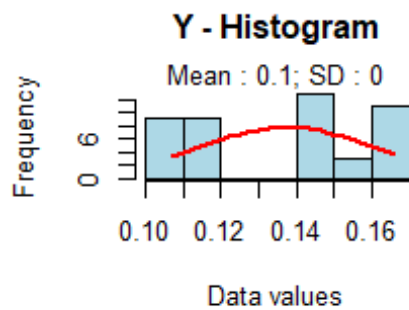
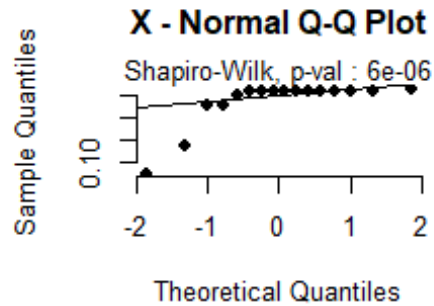
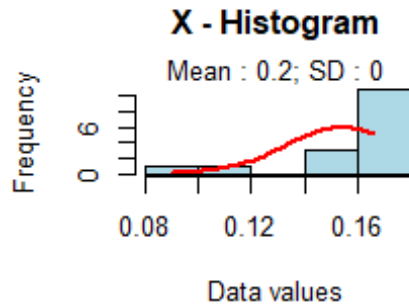
```

```
## [1] "TSPAvr"
```

```

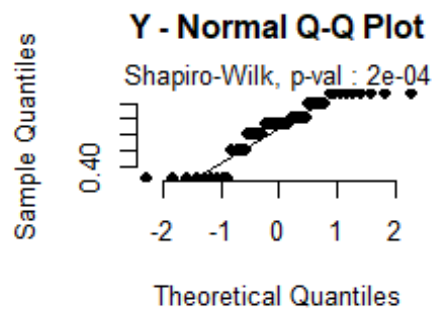
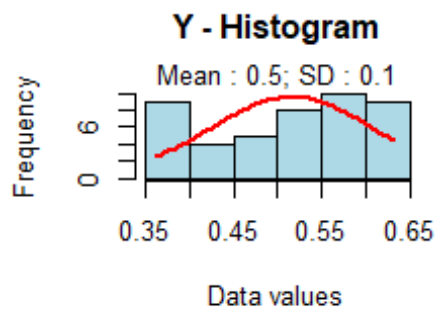
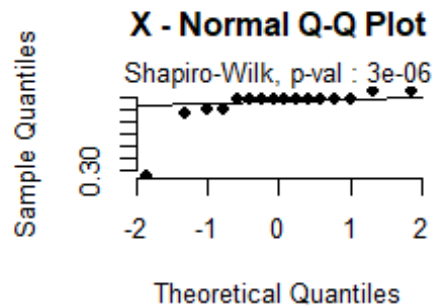
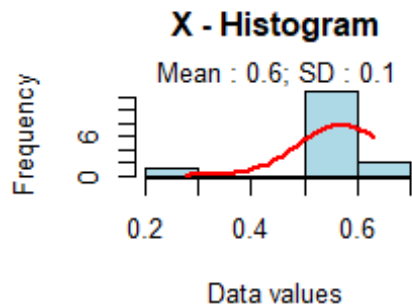
## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 6e-06 (for x) and 1e-05 (for y).
## Use a non parametric test like Wilcoxon test.

```



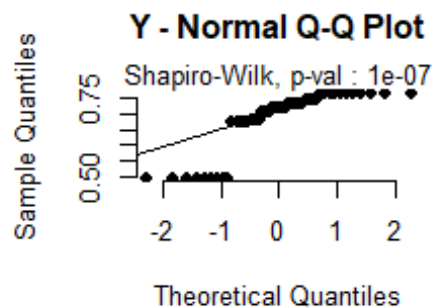
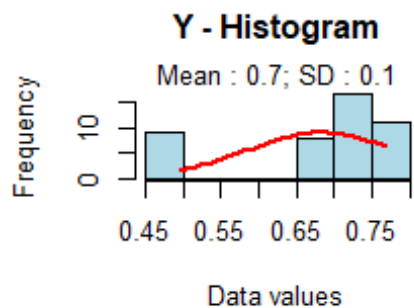
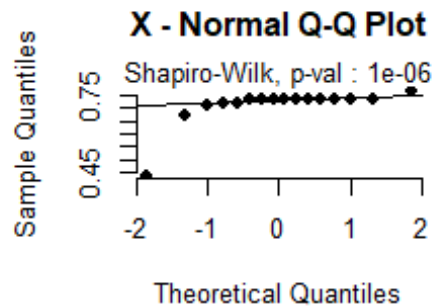
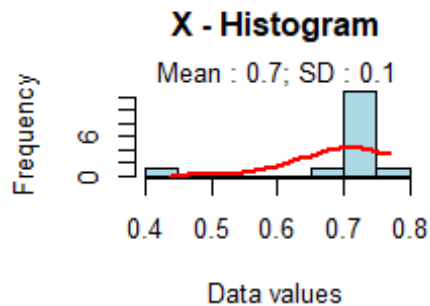
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.5346, df = 59, p-value = 0.01393
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 0.00355600 0.03022456
## sample estimates:
## mean of x mean of y
## 0.1543125 0.1374222
##
## [1] "TSPMaxAvr"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 3e-06 (for x) and 2e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



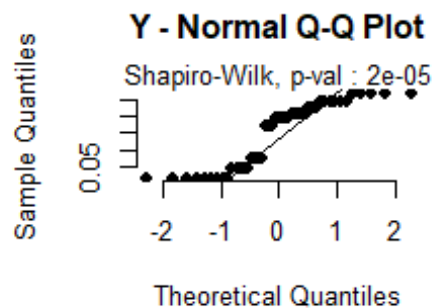
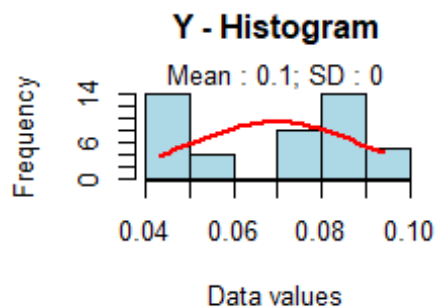
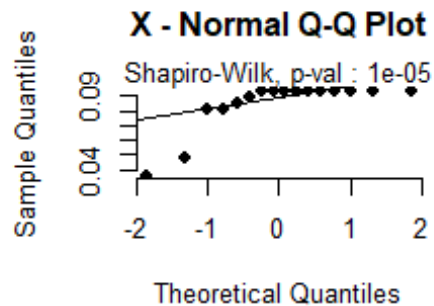
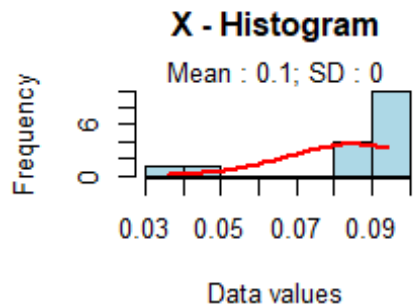
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.034, df = 59, p-value = 0.04646
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.0008773937 0.1073756618
## sample estimates:
## mean of x mean of y
## 0.5704687 0.5163422
##
## [1] "TSPMaxMax"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 1e-06 (for x) and 1e-07 (for y).
## Use a non parametric test like Wilcoxon test.
```



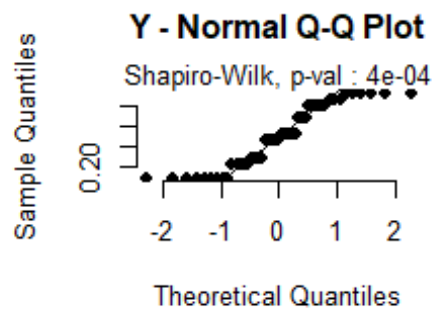
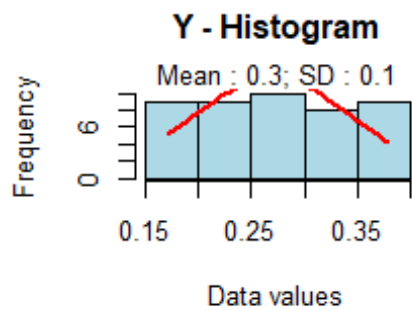
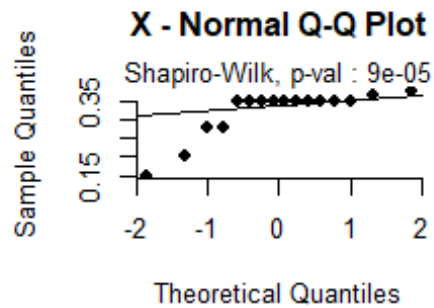
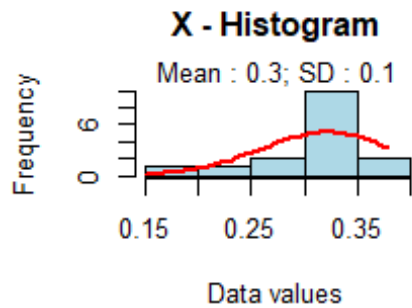
```
##
## Two Sample t-test
##
## data: x and y
## t = 1.0256, df = 59, p-value = 0.3093
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.02676603 0.08305492
## sample estimates:
## mean of x mean of y
## 0.7105000 0.6823556
##
## [1] "PM10Avr"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 1e-05 (for x) and 2e-05 (for y).
## Use a non parametric test like Wilcoxon test.
```



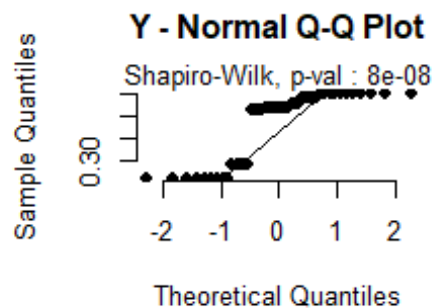
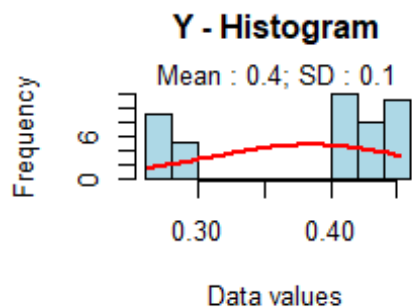
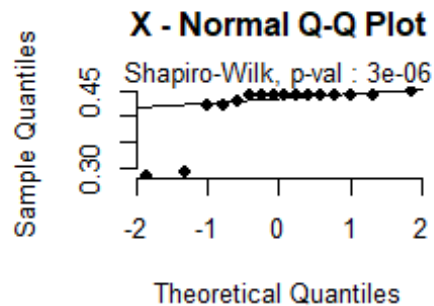
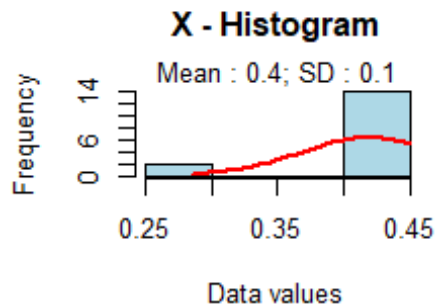
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.8021, df = 59, p-value = 0.006857
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 0.004428346 0.026549432
## sample estimates:
## mean of x mean of y
## 0.08500000 0.06951111
##
## [1] "PM10MaxAvr"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 9e-05 (for x) and 4e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```

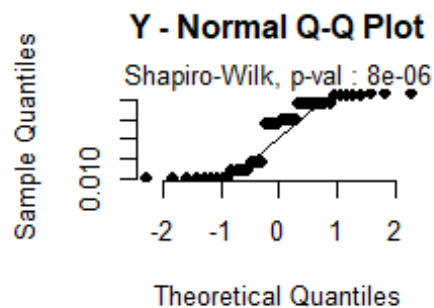
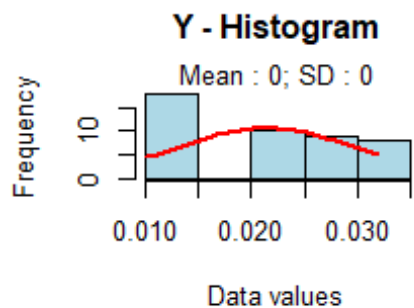
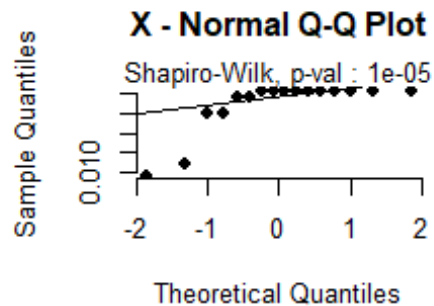
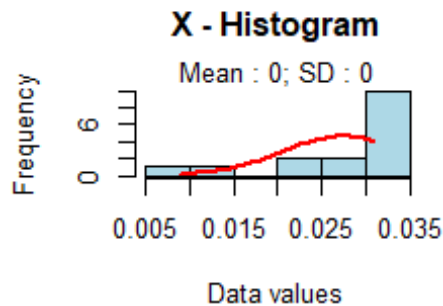
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.3587, df = 59, p-value = 0.02167
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.00763695 0.09307305
## sample estimates:
## mean of x mean of y
##  0.320975  0.270620
##
## [1] "PM10MaxMax"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 3e-06 (for x) and 8e-08 (for y).
## Use a non parametric test like Wilcoxon test.
```



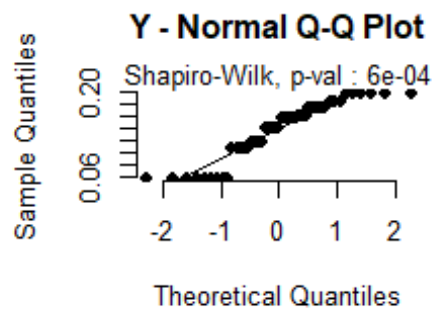
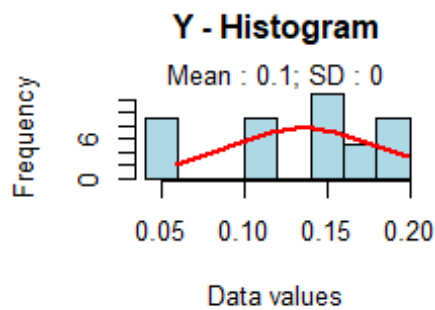
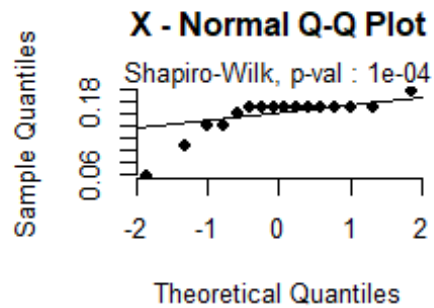
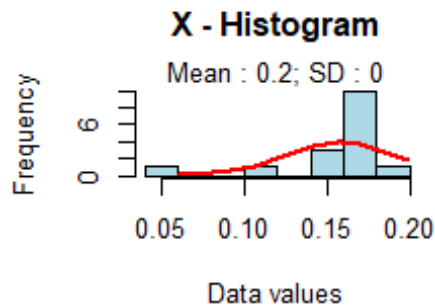
```
##
## Two Sample t-test
##
## data: x and y
## t = 1.6485, df = 59, p-value = 0.1046
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.007382729 0.076427173
## sample estimates:
## mean of x mean of y
## 0.4185000 0.3839778
##
## [1] "PM2.5Avr"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 1e-05 (for x) and 8e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```



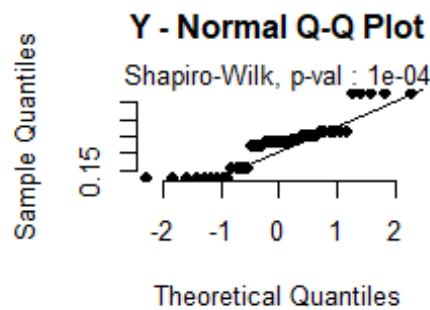
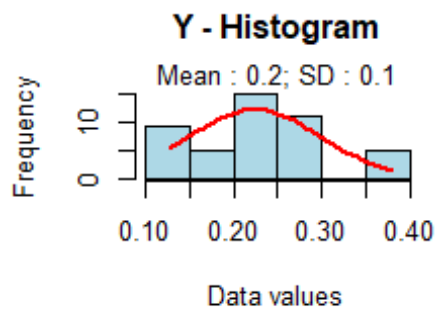
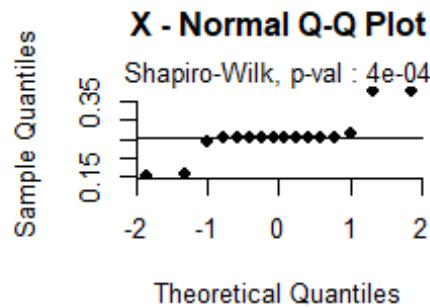
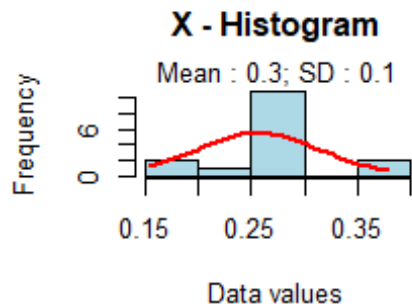
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.5291, df = 59, p-value = 0.01413
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.001260613 0.010814387
## sample estimates:
## mean of x mean of y
## 0.0274375 0.0214000
##
## [1] "PM2.5MaxAvr"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 1e-04 (for x) and 6e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



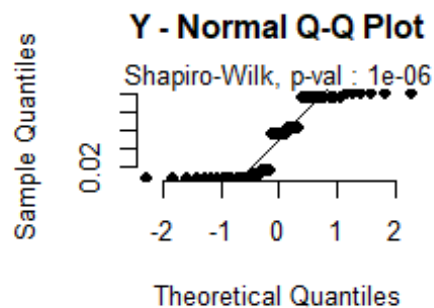
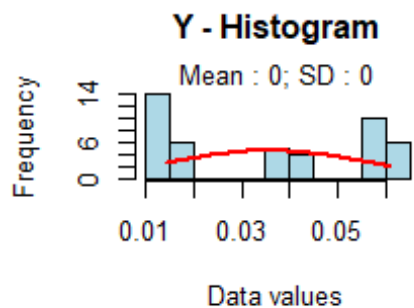
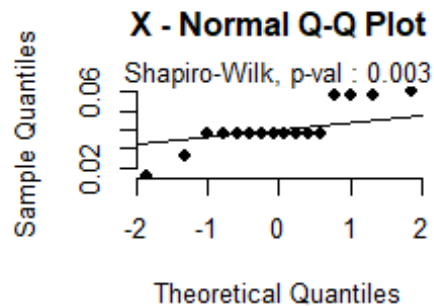
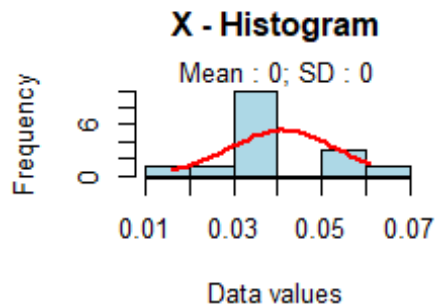
```
##
## Two Sample t-test
##
## data: x and y
## t = 1.7307, df = 59, p-value = 0.08873
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.003513487 0.048507654
## sample estimates:
## mean of x mean of y
## 0.1579437 0.1354467
##
## [1] "PM2.5MaxMax"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 4e-04 (for x) and 1e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



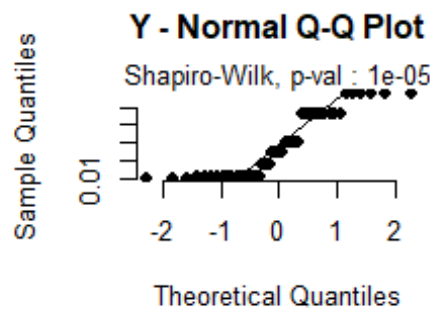
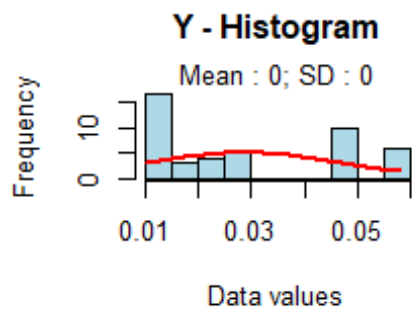
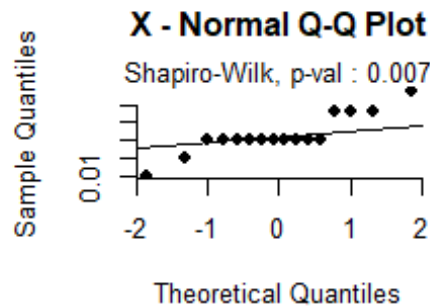
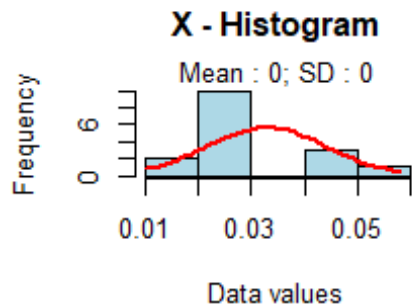
```
##
## Two Sample t-test
##
## data: x and y
## t = 1.5531, df = 59, p-value = 0.1257
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.009232888 0.073271777
## sample estimates:
## mean of x mean of y
## 0.2563750 0.2243556
##
## [1] "TSPDT"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.003 (for x) and 1e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```



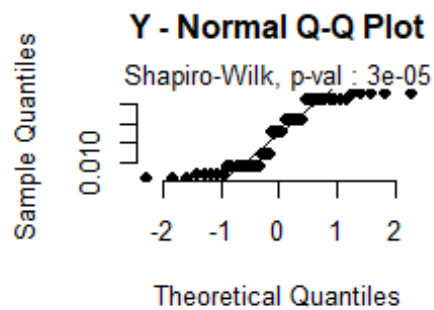
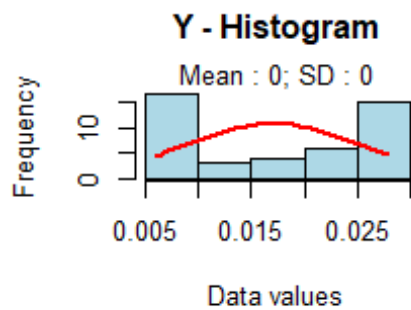
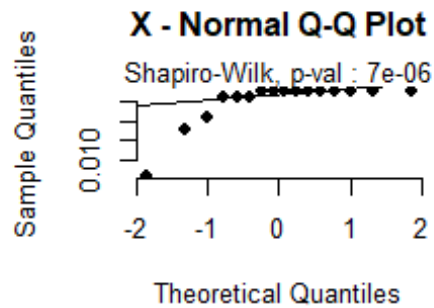
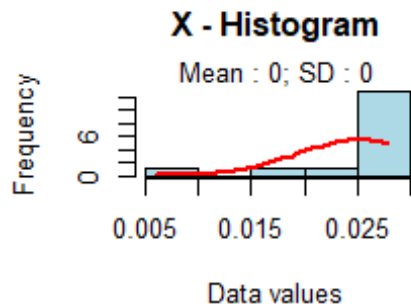
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 1.2828, df = 44.118, p-value = 0.2063
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.003118709 0.014043709
## sample estimates:
## mean of x mean of y
## 0.0410625 0.0356000
##
## [1] "PM10DT"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.007 (for x) and 1e-05 (for y).
## Use a non parametric test like Wilcoxon test.
```



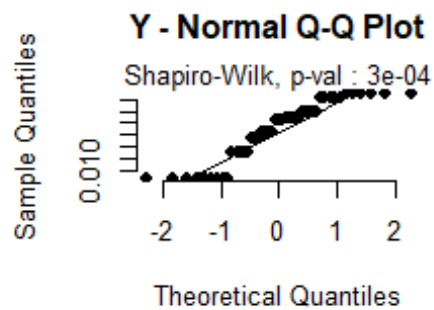
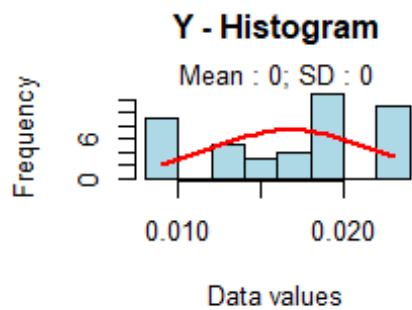
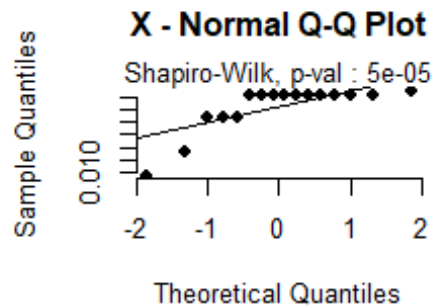
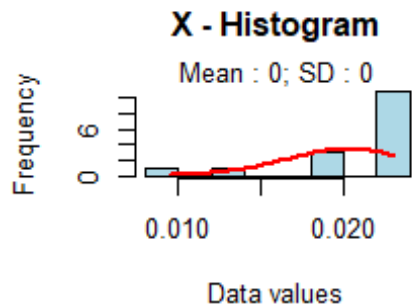
```
##
## Two Sample t-test
##
## data: x and y
## t = 0.88243, df = 59, p-value = 0.3811
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.005362589 0.013823700
## sample estimates:
## mean of x mean of y
## 0.03287500 0.02864444
##
## [1] "PM25DT"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 7e-06 (for x) and 3e-05 (for y).
## Use a non parametric test like Wilcoxon test.
```



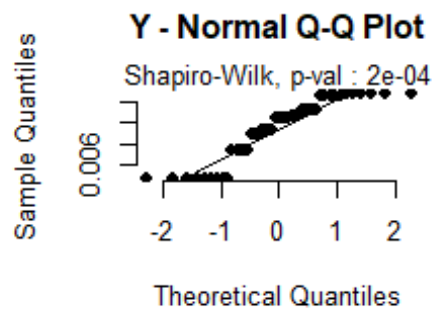
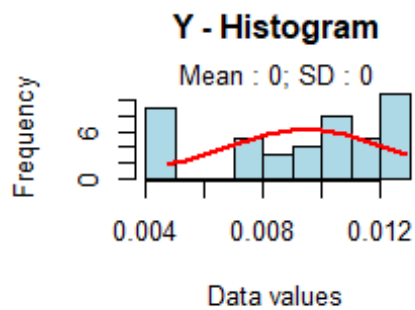
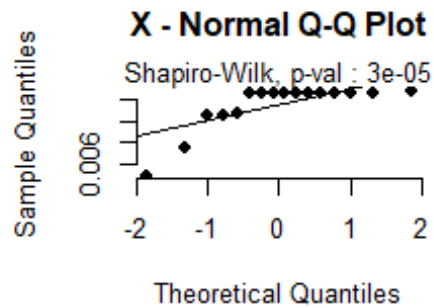
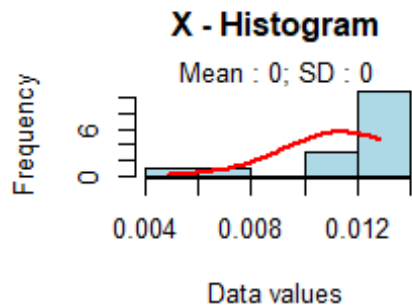
```
##
## Two Sample t-test
##
## data: x and y
## t = 3.5982, df = 59, p-value = 0.0006567
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.00360473 0.01263694
## sample estimates:
##  mean of x  mean of y
## 0.02518750 0.01706667
##
## [1] "TBPLogMax"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 5e-05 (for x) and 3e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```

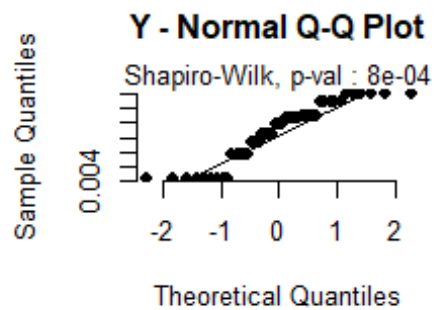
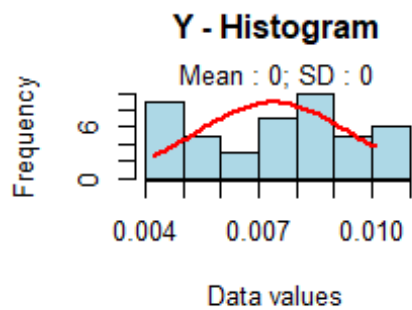
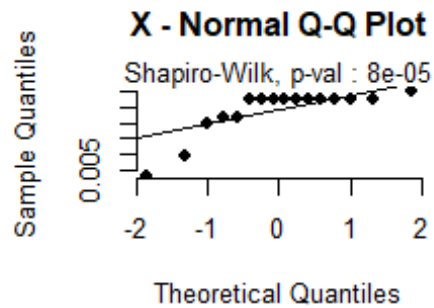
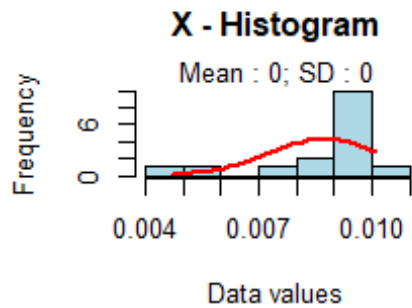
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.4891, df = 59, p-value = 0.01565
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.0006603554 0.0060746541
## sample estimates:
##  mean of x mean of y
## 0.02018250 0.01681499
##
## [1] "PLogMax"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 3e-05 (for x) and 2e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



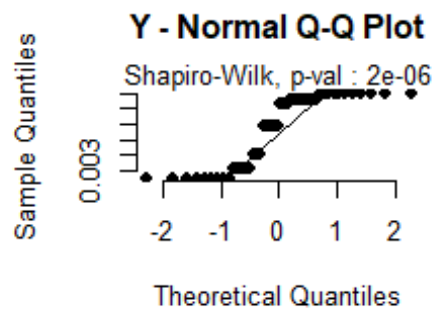
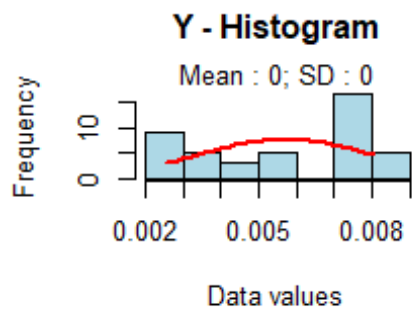
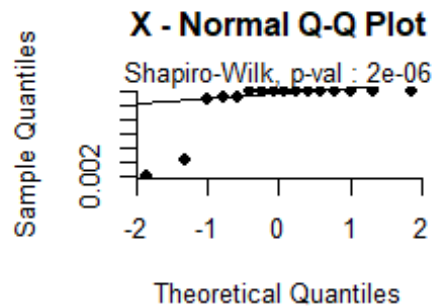
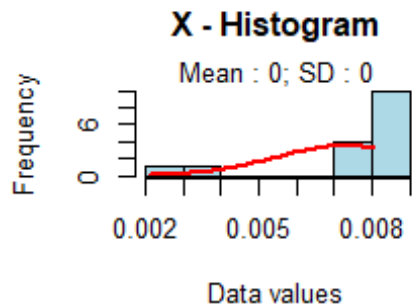
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.4873, df = 59, p-value = 0.01572
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.0003891461 0.0035912725
## sample estimates:
##  mean of x   mean of y
## 0.011430226 0.009440017
##
## [1] "TBlogMax"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 8e-05 (for x) and 8e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



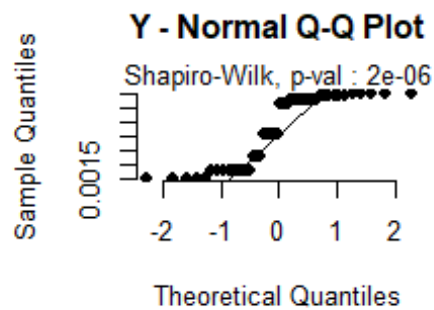
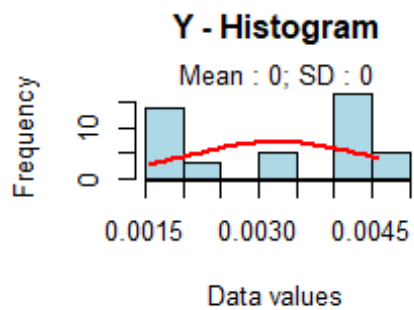
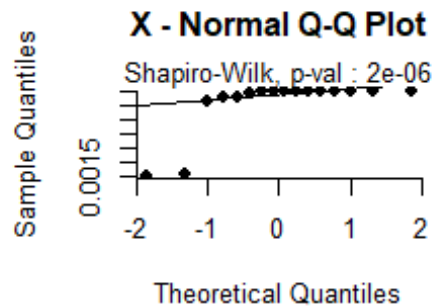
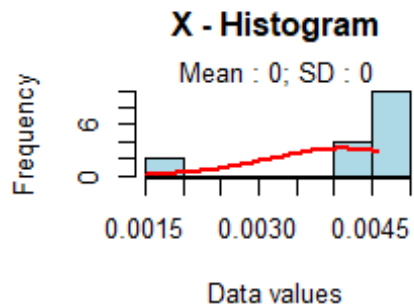
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.4845, df = 59, p-value = 0.01583
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 0.0002680315 0.0024865602
## sample estimates:
## mean of x mean of y
## 0.008752270 0.007374975
##
## [1] "TBPLogAvr"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 2e-06 (for x) and 2e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```



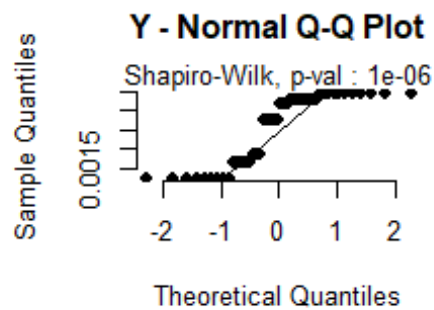
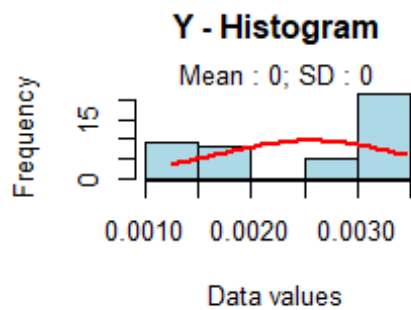
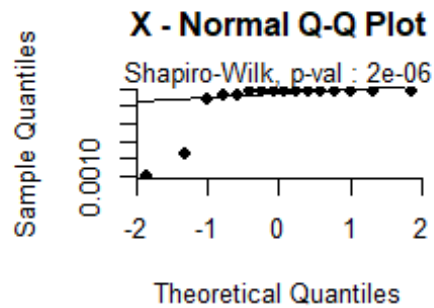
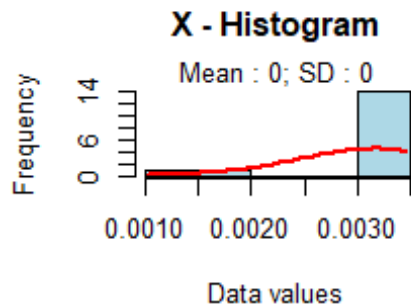
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.4524, df = 58, p-value = 0.01722
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.000285556 0.002822282
## sample estimates:
##  mean of x   mean of y
## 0.007256441 0.005702521
##
## [1] "PLogAvr"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 2e-06 (for x) and 2e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```



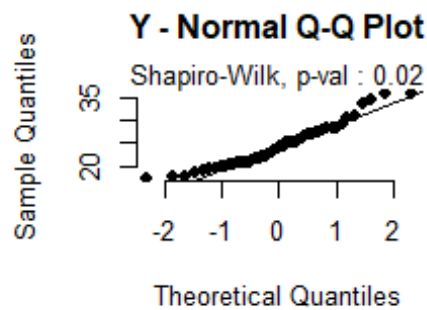
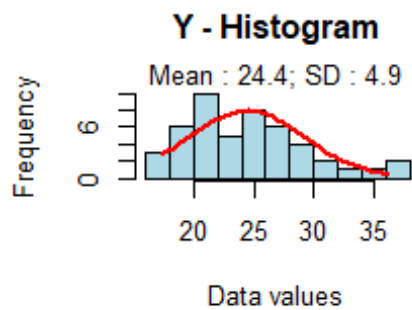
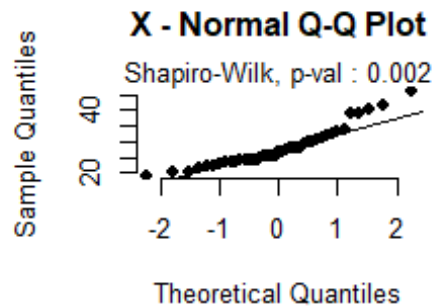
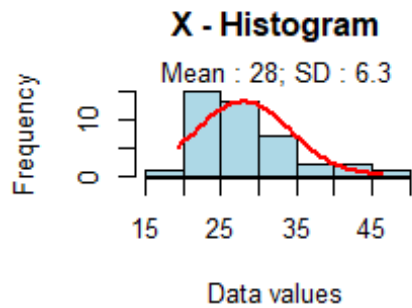
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.5521, df = 58, p-value = 0.01336
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.0001892584 0.0015658986
## sample estimates:
##  mean of x   mean of y
## 0.004121192 0.003243613
##
## [1] "TBLLogAvr"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 2e-06 (for x) and 1e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```



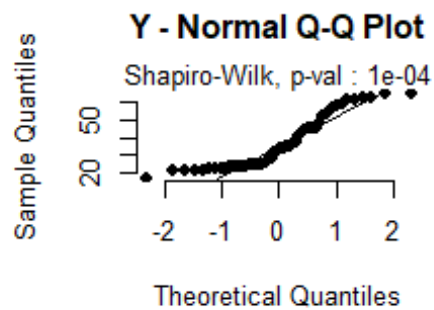
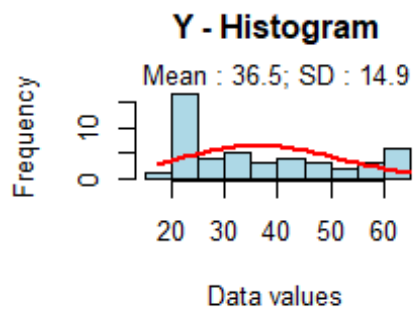
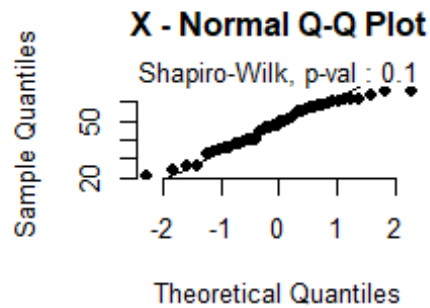
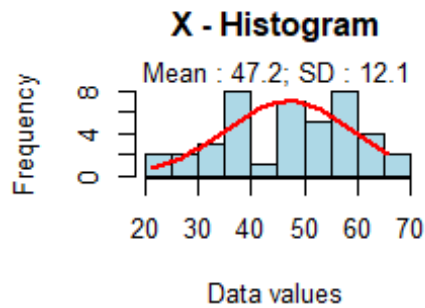
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.3424, df = 58, p-value = 0.02261
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 8.689459e-05 1.107902e-03
## sample estimates:
## mean of x mean of y
## 0.003155280 0.002557881
##
## [1] "BMI"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.002 (for x) and 0.02 (for y).
## Use a non parametric test like Wilcoxon test.
```

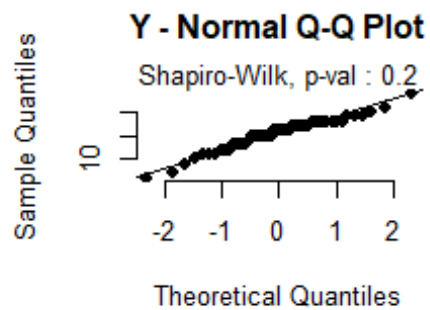
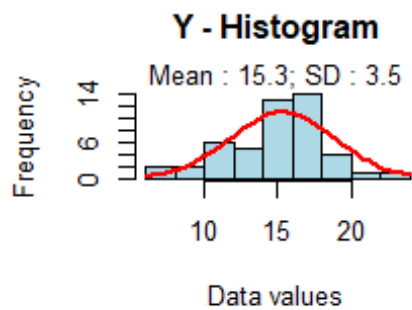
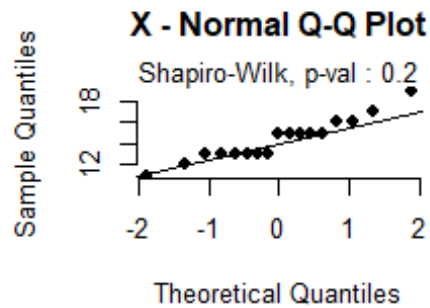
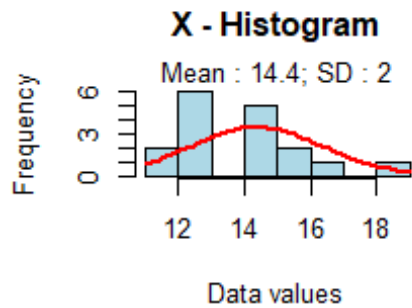


```
##
## Two Sample t-test
##
## data: x and y
## t = 3.0063, df = 87, p-value = 0.003456
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.203632 5.900535
## sample estimates:
## mean of x mean of y
## 27.96000 24.40792
##
## [1] "Age"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.1 (for x) and 1e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```

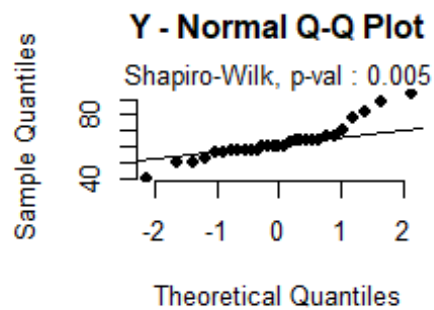
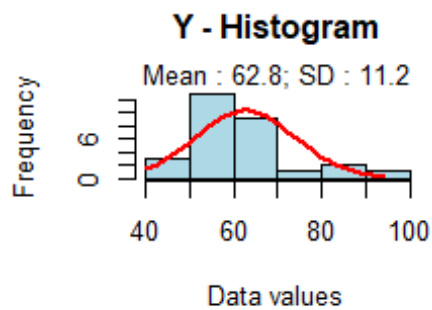
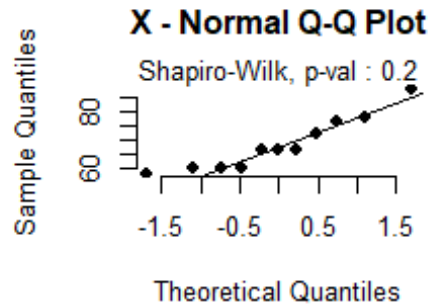
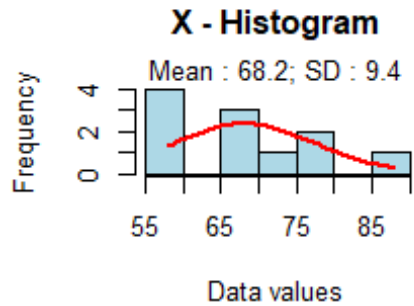


```
##
## Two Sample t-test
##
## data: x and y
## t = 3.7293, df = 88, p-value = 0.0003393
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  5.014129 16.454233
## sample estimates:
## mean of x mean of y
## 47.22733 36.49315
##
## [1] "EducYrs"
```

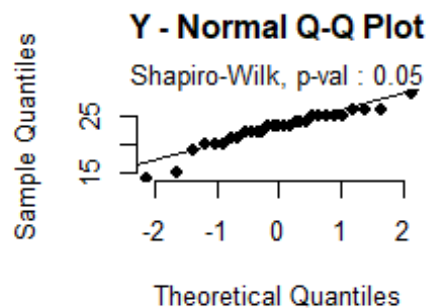
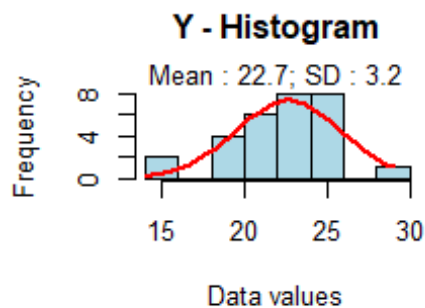
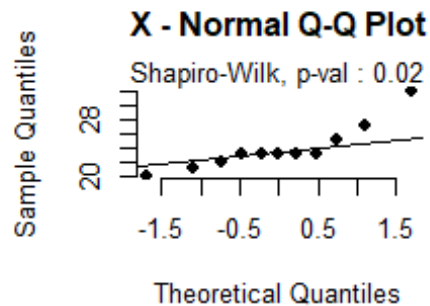
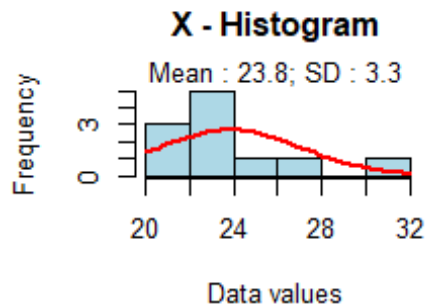
```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -1.3948, df = 50.048, p-value = 0.1692
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.392109  0.431325
## sample estimates:
## mean of x mean of y
##  14.35294  15.33333
##
## [1] "DL"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.2 (for x) and 0.005 (for y).
## Use a non parametric test like Wilcoxon test.
```



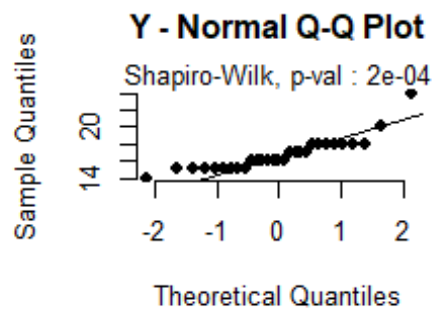
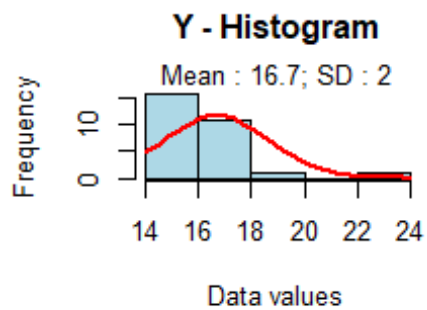
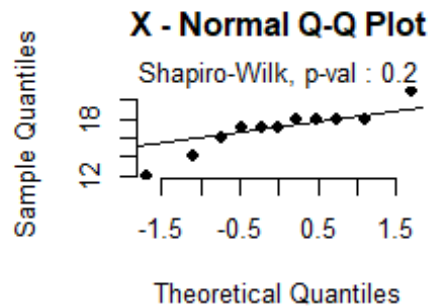
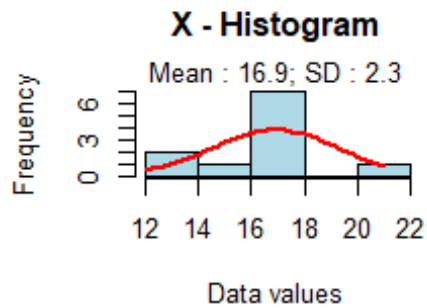
```
##
## Two Sample t-test
##
## data: x and y
## t = 1.426, df = 38, p-value = 0.162
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -2.275567 13.121962
## sample estimates:
## mean of x mean of y
## 68.18182 62.75862
##
## [1] "JD"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.02 (for x) and 0.05 (for y).
## Use a non parametric test like Wilcoxon test.
```

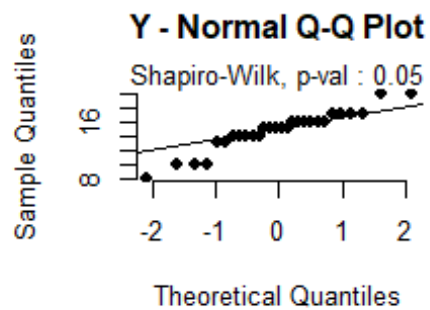
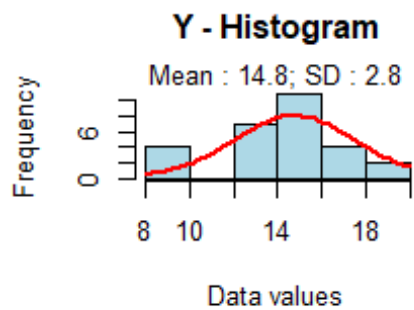
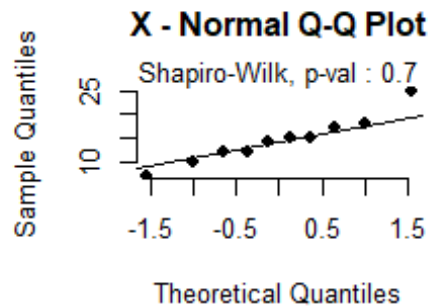
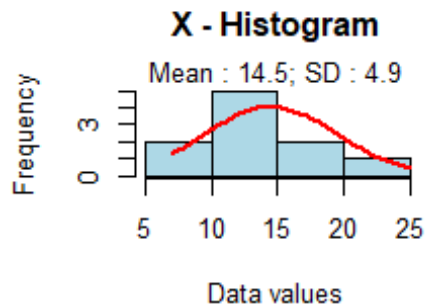


```
##
## Two Sample t-test
##
## data: x and y
## t = 1.025, df = 38, p-value = 0.3119
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.134064 3.460083
## sample estimates:
## mean of x mean of y
## 23.81818 22.65517
##
## [1] "CS"

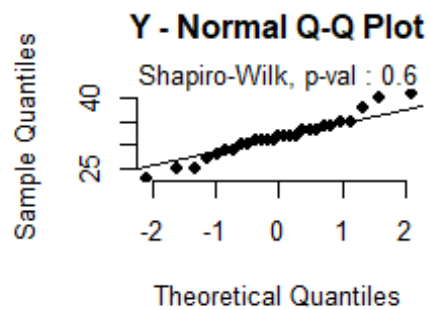
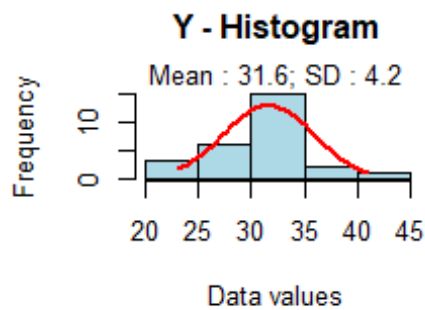
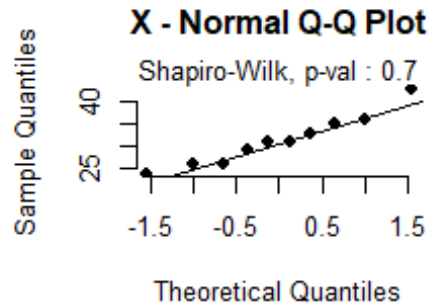
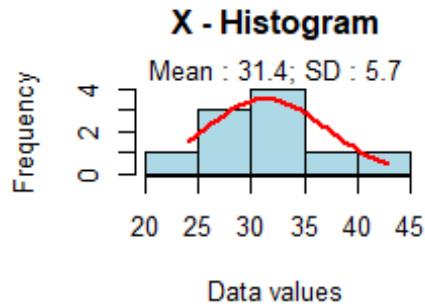
## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.2 (for x) and 2e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



```
##
## Two Sample t-test
##
## data: x and y
## t = 0.29728, df = 38, p-value = 0.7679
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.274846 1.713717
## sample estimates:
## mean of x mean of y
## 16.90909 16.68966
##
## [1] "SS"
```

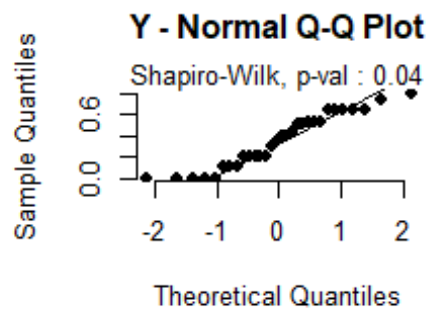
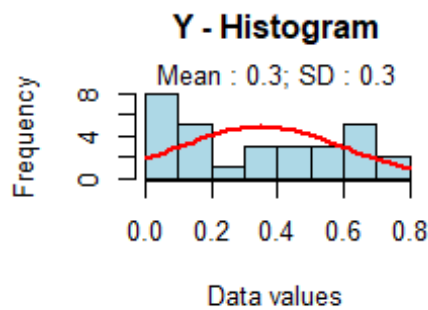
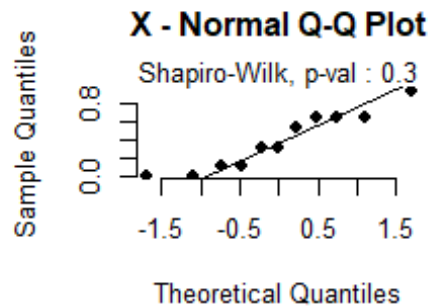
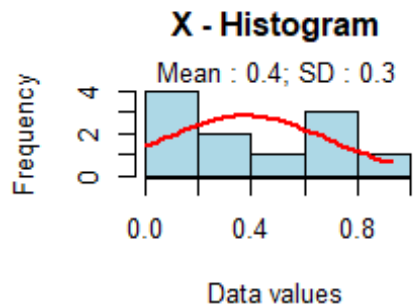


```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -0.15209, df = 11.1, p-value = 0.8818
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.863895  3.363895
## sample estimates:
## mean of x mean of y
##    14.50    14.75
##
## [1] "SocSup"
```



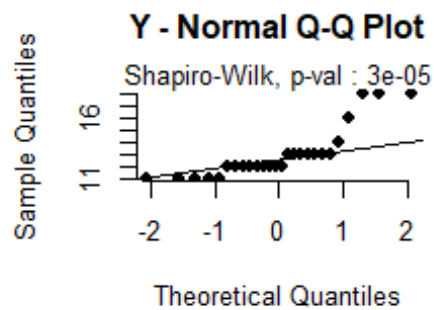
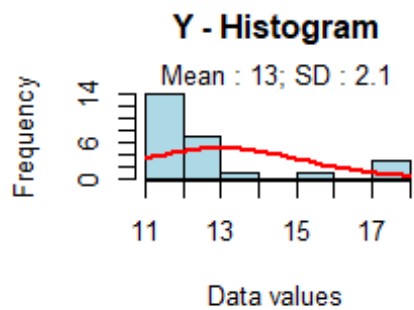
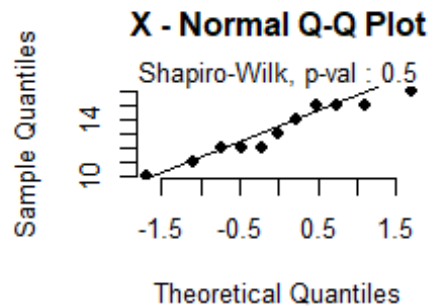
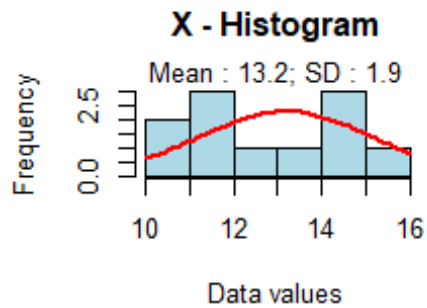
```
##
## Two Sample t-test
##
## data: x and y
## t = -0.1343, df = 35, p-value = 0.8939
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -3.700801 3.241542
## sample estimates:
## mean of x mean of y
## 31.40000 31.62963
##
## [1] "JobDis"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.3 (for x) and 0.04 (for y).
## Use a non parametric test like Wilcoxon test.
```

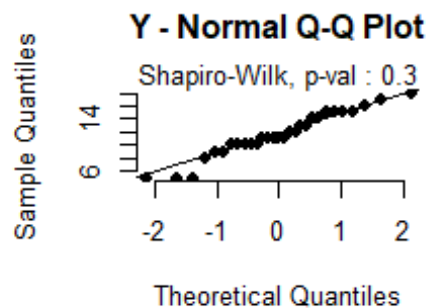
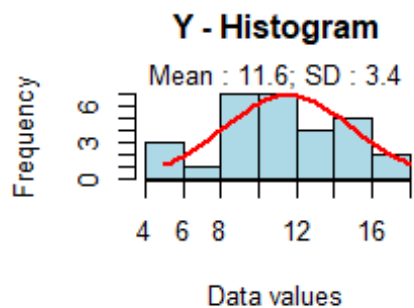
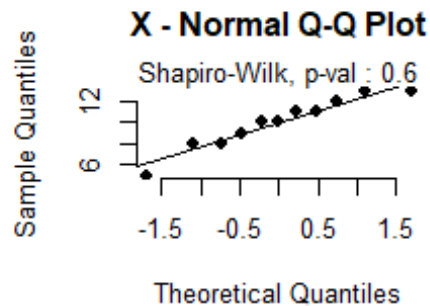
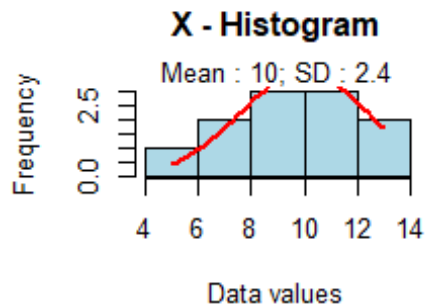


```
##
## Two Sample t-test
##
## data: x and y
## t = 0.31632, df = 39, p-value = 0.7534
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.1612873 0.2210852
## sample estimates:
## mean of x mean of y
## 0.3787879 0.3488889
##
## [1] "JI"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.5 (for x) and 3e-05 (for y).
## Use a non parametric test like Wilcoxon test.
```

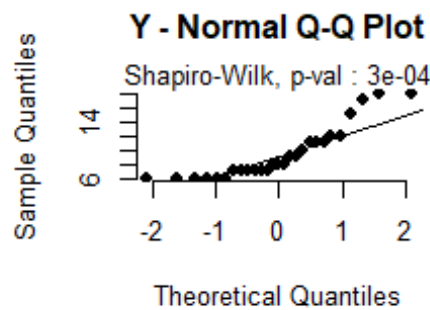
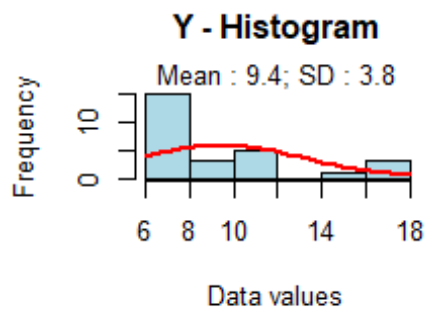
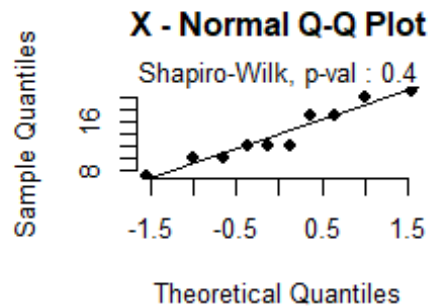
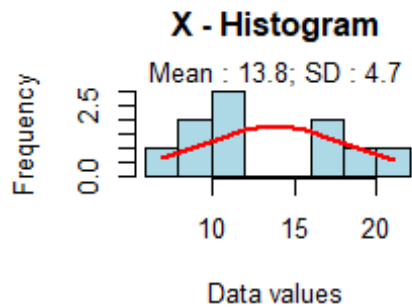


```
##
## Two Sample t-test
##
## data: x and y
## t = 0.24286, df = 35, p-value = 0.8095
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.338045 1.701682
## sample estimates:
## mean of x mean of y
## 13.18182 13.00000
##
## [1] "PE"
```

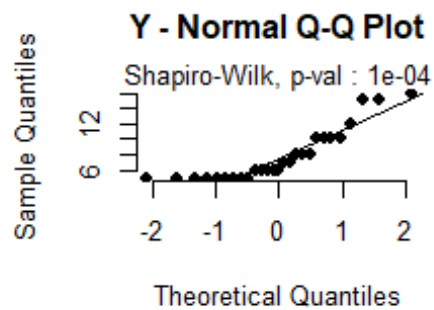
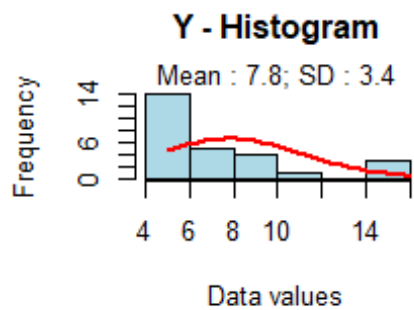
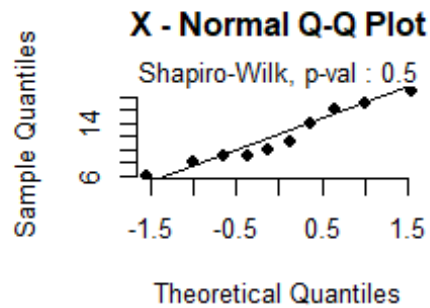
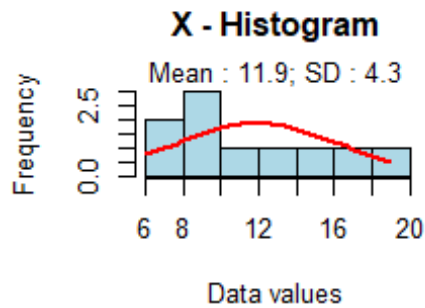
```
##
## Two Sample t-test
##
## data: x and y
## t = -1.3836, df = 38, p-value = 0.1746
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -3.8221047 0.7186564
## sample estimates:
## mean of x mean of y
## 10.00000 11.55172
##
## [1] "ERI_Ef_Ph"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.4 (for x) and 3e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```

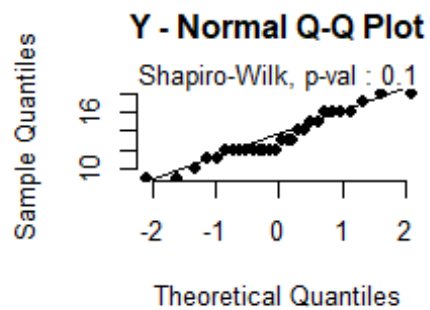
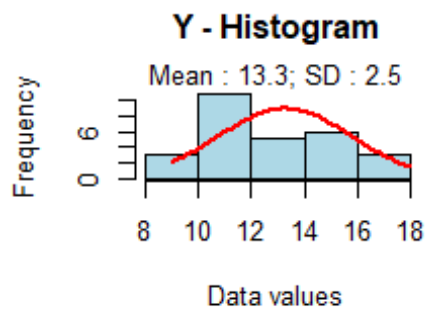
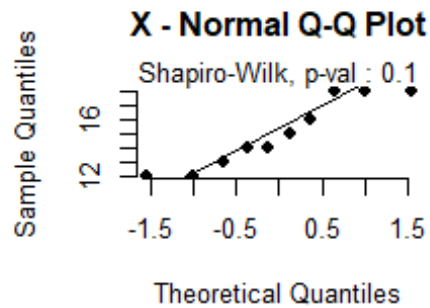
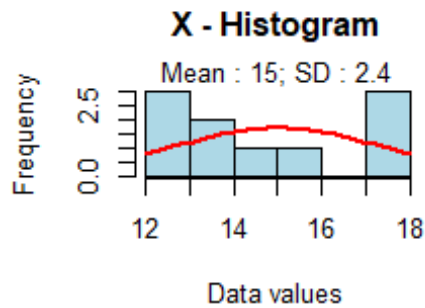


```
##
## Two Sample t-test
##
## data: x and y
## t = 2.9303, df = 35, p-value = 0.005927
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.338062 7.373049
## sample estimates:
## mean of x mean of y
## 13.800000  9.444444
##
## [1] "ERI_Ef_M"

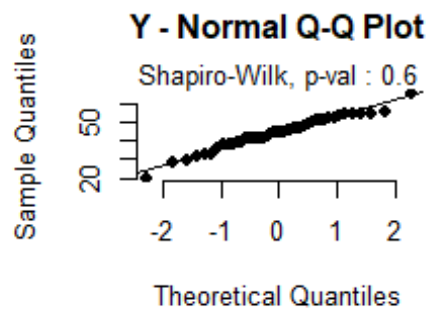
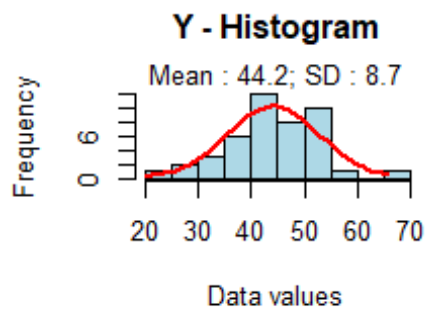
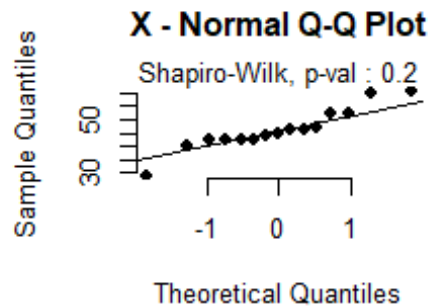
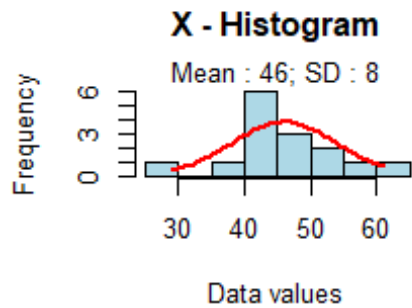
## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.5 (for x) and 1e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



```
##
## Two Sample t-test
##
## data: x and y
## t = 3.0277, df = 35, p-value = 0.004604
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.346019 6.824352
## sample estimates:
## mean of x mean of y
## 11.900000 7.814815
##
## [1] "Overcom"
```

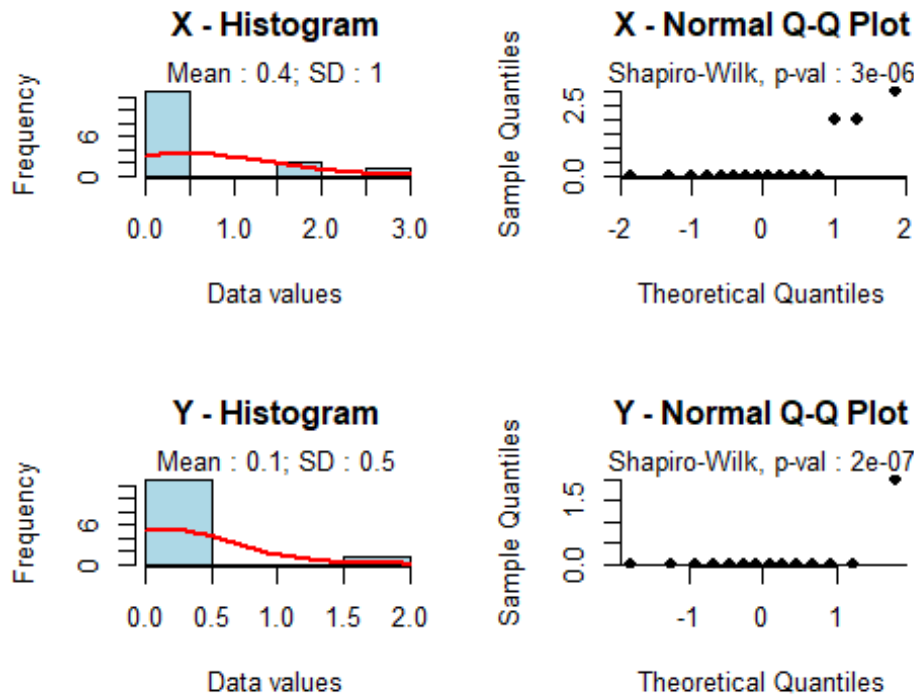


```
##
## Two Sample t-test
##
## data: x and y
## t = 1.8836, df = 36, p-value = 0.06771
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.1314893 3.5600607
## sample estimates:
## mean of x mean of y
## 15.00000 13.28571
##
## [1] "SAnx"
```



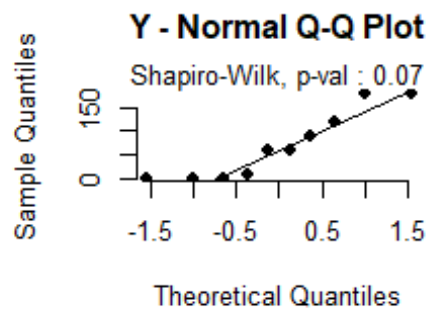
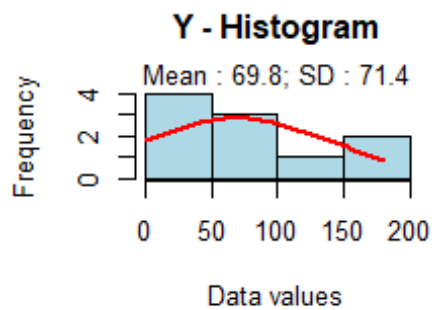
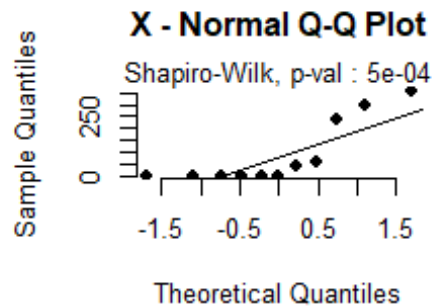
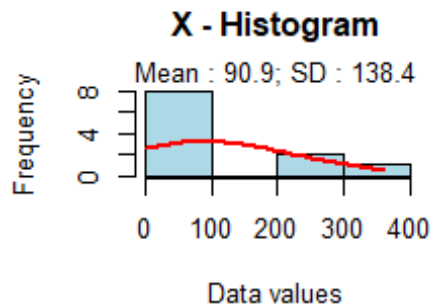
```
##
## Two Sample t-test
##
## data: x and y
## t = 0.72108, df = 57, p-value = 0.4738
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -3.271357 6.953175
## sample estimates:
## mean of x mean of y
## 46.00000 44.15909
##
## [1] "PasSMK"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 3e-06 (for x) and 2e-07 (for y).
## Use a non parametric test like Wilcoxon test.
```



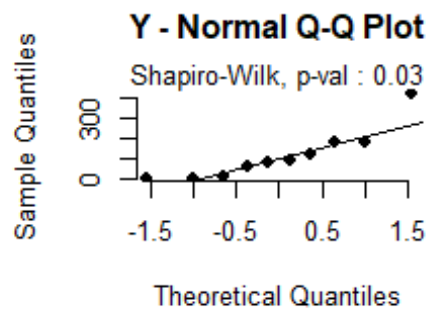
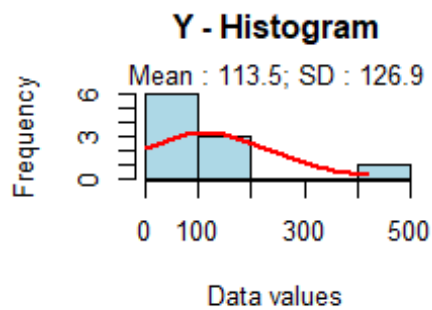
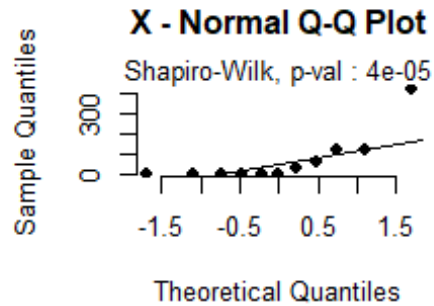
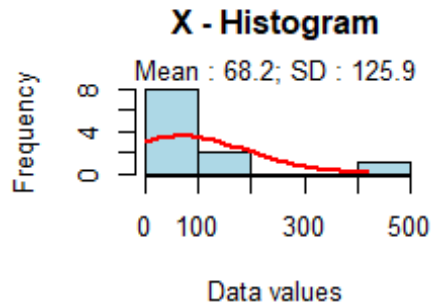
```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = 1.0518, df = 23.978, p-value = 0.3034
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2835752  0.8728609
## sample estimates:
## mean of x mean of y
## 0.4375000 0.1428571
##
## [1] "PA_RigMin"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 5e-04 (for x) and 0.07 (for y).
## Use a non parametric test like Wilcoxon test.
```



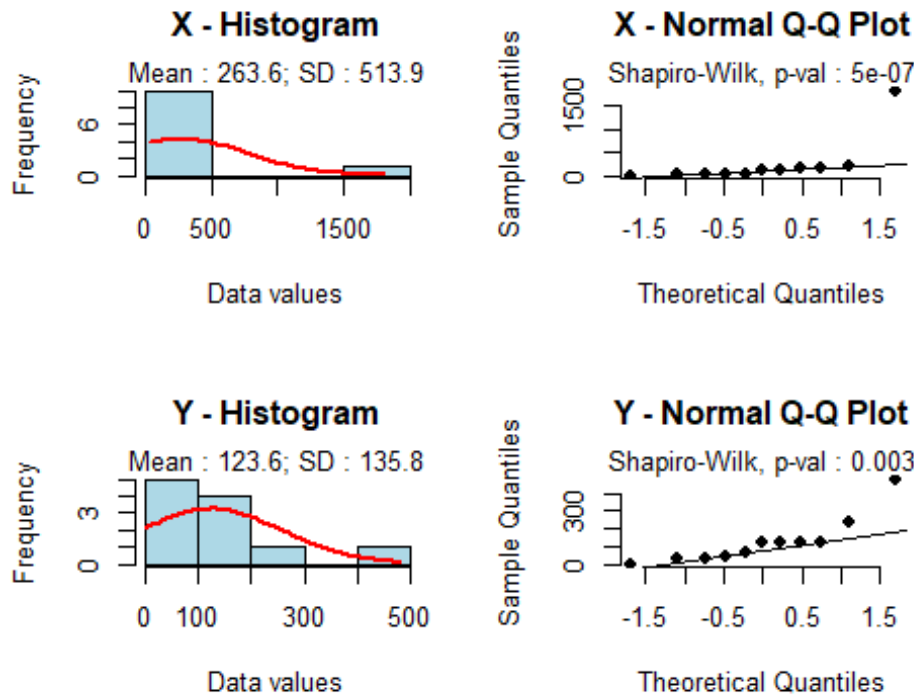
```
##
## Two Sample t-test
##
## data: x and y
## t = 0.43227, df = 19, p-value = 0.6704
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -81.10039 123.31857
## sample estimates:
## mean of x mean of y
## 90.90909 69.80000
##
## [1] "PA_ModMin"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 4e-05 (for x) and 0.03 (for y).
## Use a non parametric test like Wilcoxon test.
```



```
##
## Two Sample t-test
##
## data: x and y
## t = -0.82069, df = 19, p-value = 0.422
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -160.89370 70.25733
## sample estimates:
## mean of x mean of y
## 68.18182 113.50000
##
## [1] "Walk_Min"

## Warning in rquery.t.test(exp_n.BA11[[i]], exp_n.BA01[[i]]): x or y is not
normally distributed : Shapiro test p-value : 5e-07 (for x) and 0.003 (for y).
## Use a non parametric test like Wilcoxon test.
```

```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = 0.87356, df = 11.39, p-value = 0.4004
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -211.2707  491.2707
## sample estimates:
## mean of x mean of y
##  263.6364  123.6364

#wilcox-mecm
for (i in 1:num1) {
  print(names(exp_n1[i]))
  print(wilcox.test(exp_n.BA11[[i]], exp_n.BA01[[i]]), alternative = "g")
}

## [1] "TSPAвр"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
## compute
## exact p-value with ties

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 508, p-value = 0.01442
## alternative hypothesis: true location shift is not equal to 0
```

```

##
## [1] "TSPMaxAvr"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 498, p-value = 0.02256
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TSPMaxMax"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 389.5, p-value = 0.6295
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM10Avr"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 563, p-value = 0.0007822
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM10MaxAvr"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 477, p-value = 0.0533
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM10MaxMax"

```

```

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 444, p-value = 0.1654
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM2.5Avr"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 522.5, p-value = 0.007085
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM2.5MaxAvr"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 468, p-value = 0.07456
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM2.5MaxMax"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 490, p-value = 0.03157
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TSPDT"

```

```

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 404.5, p-value = 0.4619
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM10DT"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 430.5, p-value = 0.2418
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM25DT"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 580.5, p-value = 0.0002489
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TBPLogMax"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 503, p-value = 0.01809
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PLogMax"

```

```

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 502, p-value = 0.01892
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TBLogMax"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 503, p-value = 0.01809
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TBPLogAvr"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 544, p-value = 0.001174
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PLogAvr"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 535, p-value = 0.001984
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TBLogAvr"

```

```

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 544, p-value = 0.001174
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "BMI"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 1318, p-value = 0.006049
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "Age"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 1438, p-value = 0.0005133
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "EducYrs"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 303, p-value = 0.1159
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "DL"

```

```

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 221, p-value = 0.0626
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "JD"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 173.5, p-value = 0.6791
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "CS"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 189, p-value = 0.3695
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "SS"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 124, p-value = 0.6042
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "SocSup"

```

```

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 127.5, p-value = 0.8101
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "JobDis"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 173, p-value = 0.8239
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "JI"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 158.5, p-value = 0.6092
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PE"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 112.5, p-value = 0.1559
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "ERI_Ef_Ph"

```



```

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 213, p-value = 0.007564
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "ERI_Ef_M"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 215.5, p-value = 0.005655
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "Overcom"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 194.5, p-value = 0.06895
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "SANx"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 365.5, p-value = 0.5416
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PasSMK"

```

```

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 125.5, p-value = 0.3599
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PA_RigMin"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 48, p-value = 0.633
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PA_ModMin"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 35, p-value = 0.1569
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "Walk_Min"

## Warning in wilcox.test.default(exp_n.BA11[[i]], exp_n.BA01[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA11[[i]] and exp_n.BA01[[i]]
## W = 72, p-value = 0.4636
## alternative hypothesis: true location shift is not equal to 0

#T-mecm
for (i in 1:num2) {
  print(names(exp_n2[i]))
}

```

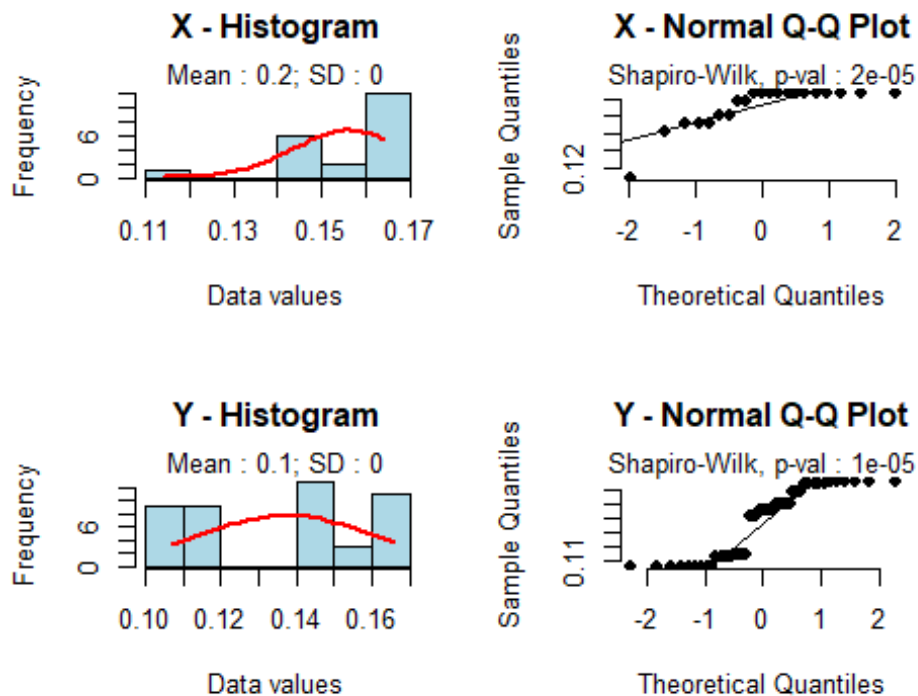
```

print(rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]), alternative = "g")
}

## [1] "TSPAвр"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 2e-05 (for x) and 1e-05 (for y).
## Use a non parametric test like Wilcoxon test.

```

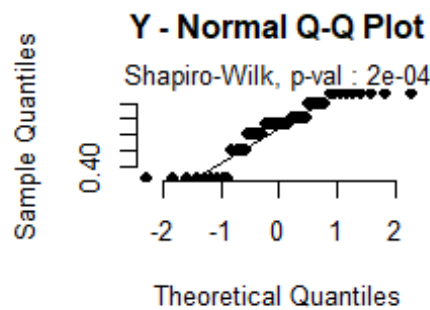
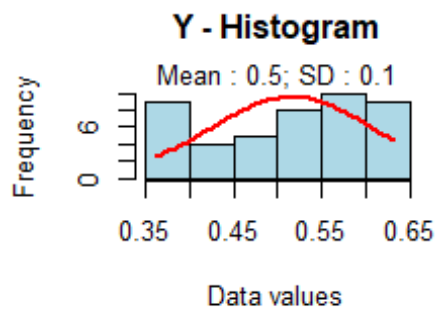
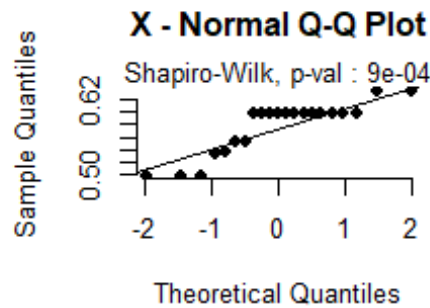
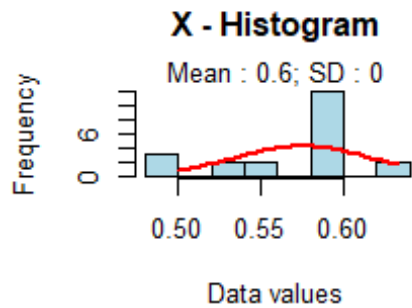


```

##
## Welch Two Sample t-test
##
## data: x and y
## t = 4.1773, df = 62.413, p-value = 9.334e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.00968887 0.02746669
## sample estimates:
## mean of x mean of y
## 0.1560000 0.1374222
##
## [1] "TSPMaxAvr"

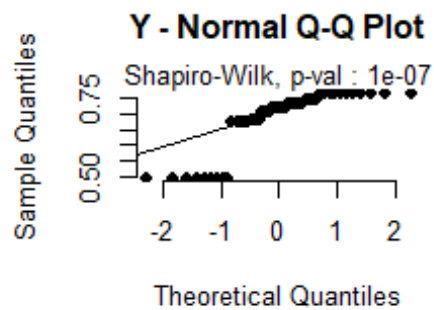
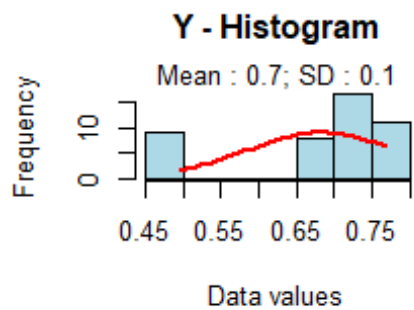
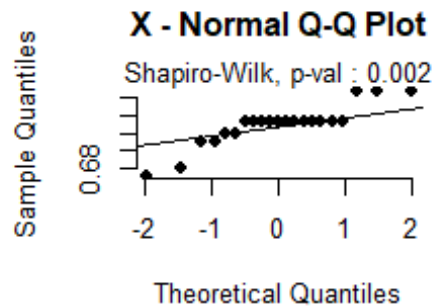
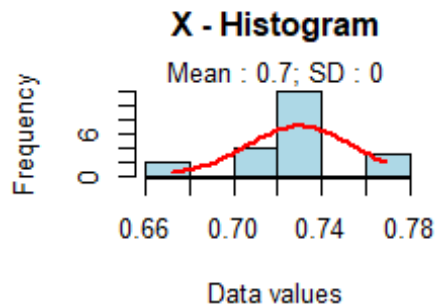
## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 9e-04 (for x) and 2e-04 (for y).
## Use a non parametric test like Wilcoxon test.

```



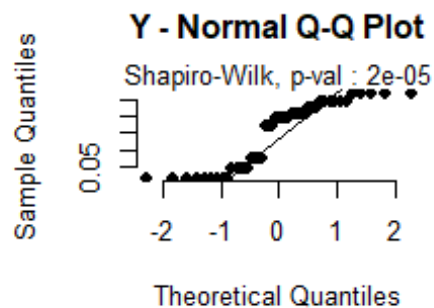
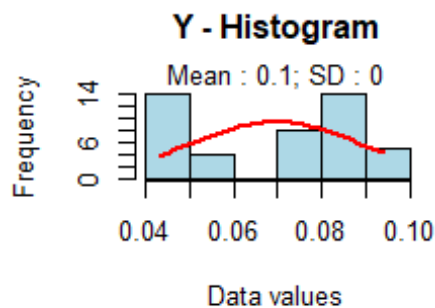
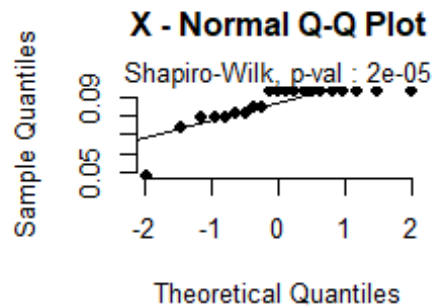
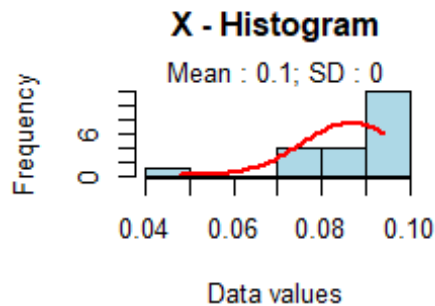
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.5526, df = 63.765, p-value = 0.0007239
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.02580984 0.09214381
## sample estimates:
## mean of x mean of y
## 0.5753190 0.5163422
##
## [1] "TSPMaxMax"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.002 (for x) and 1e-07 (for y).
## Use a non parametric test like Wilcoxon test.
```



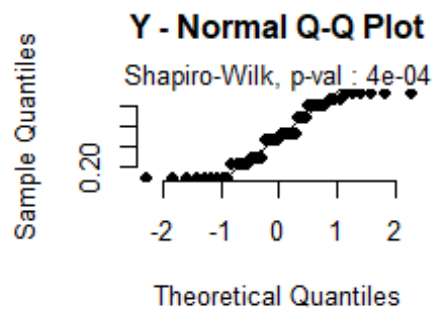
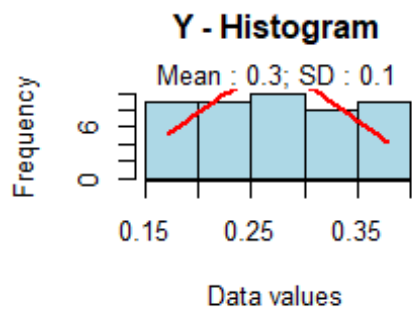
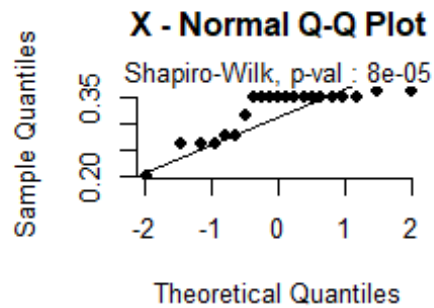
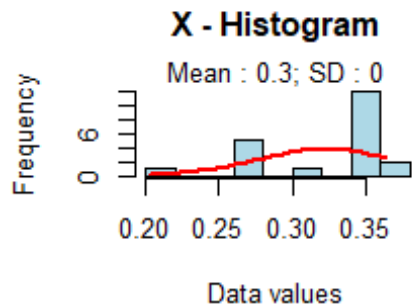
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.008, df = 53.821, p-value = 0.003993
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.01583848 0.07916469
## sample estimates:
## mean of x mean of y
## 0.7298571 0.6823556
##
## [1] "PM10Avr"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 2e-05 (for x) and 2e-05 (for y).
## Use a non parametric test like Wilcoxon test.
```



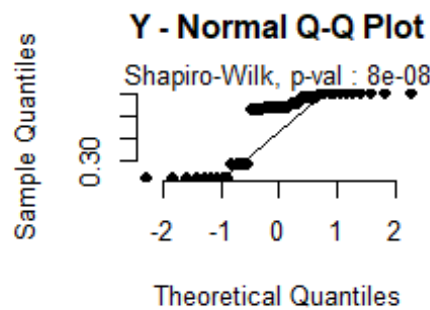
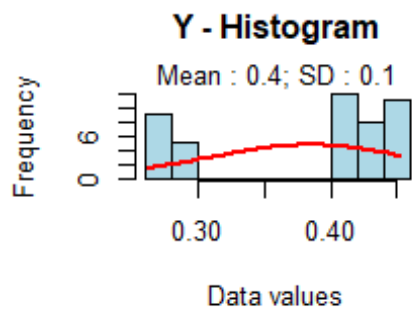
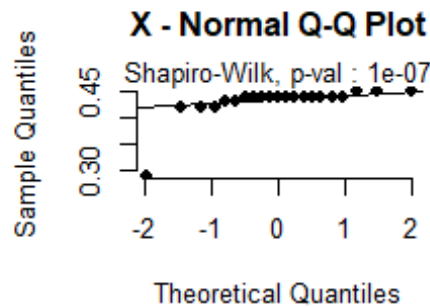
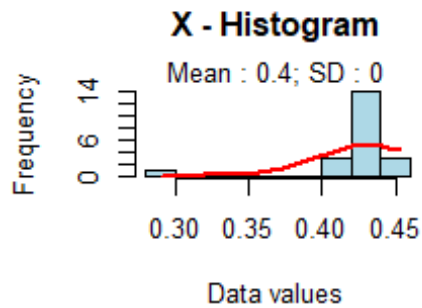
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 4.497, df = 60.791, p-value = 3.161e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.00950019 0.02471568
## sample estimates:
## mean of x mean of y
## 0.08661905 0.06951111
##
## [1] "PM10MaxAvr"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 8e-05 (for x) and 4e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



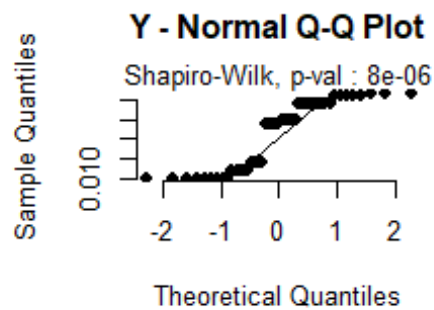
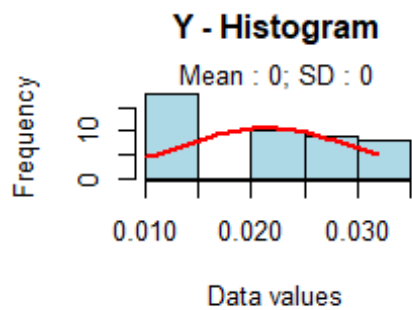
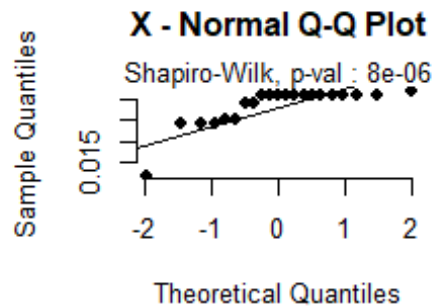
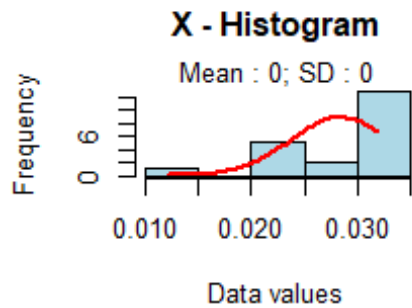
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.4238, df = 60.221, p-value = 0.001116
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.02148137 0.08184054
## sample estimates:
## mean of x mean of y
##  0.322281  0.270620
##
## [1] "PM10MaxMax"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 1e-07 (for x) and 8e-08 (for y).
## Use a non parametric test like Wilcoxon test.
```



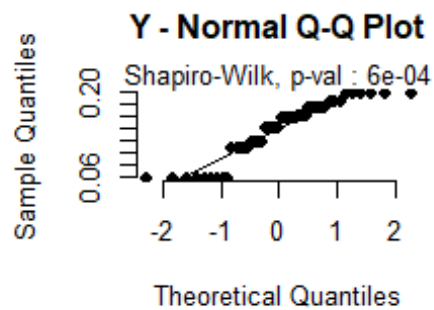
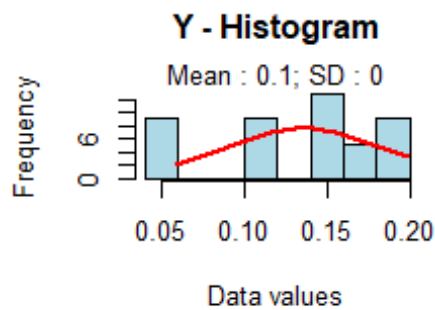
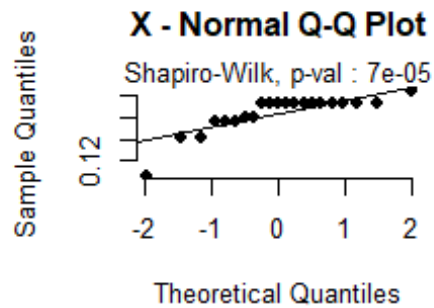
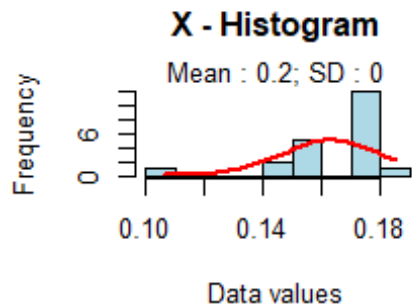
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.4311, df = 63.766, p-value = 0.00106
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.01962187 0.07432734
## sample estimates:
## mean of x mean of y
## 0.4309524 0.3839778
##
## [1] "PM2.5Avr"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 8e-06 (for x) and 8e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```

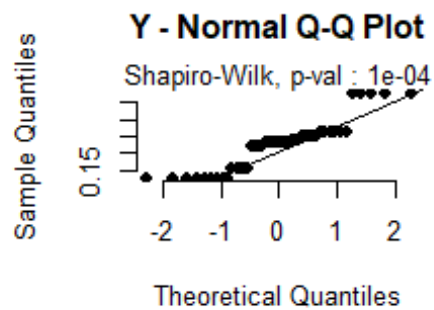
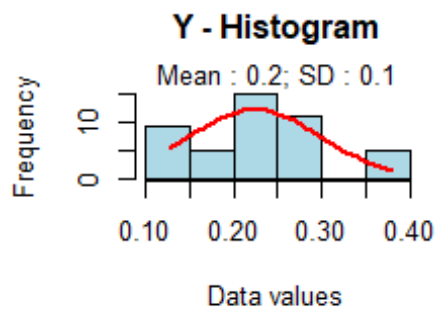
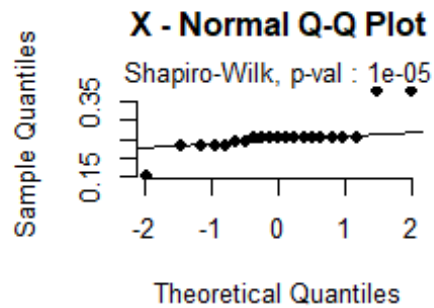
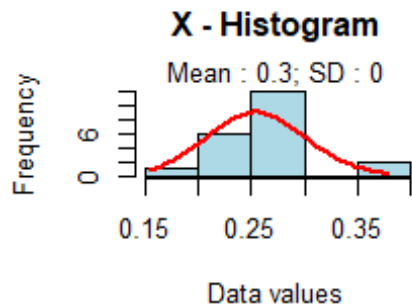
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 4.2491, df = 62.149, p-value = 7.321e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.003696945 0.010264959
## sample estimates:
## mean of x mean of y
## 0.02838095 0.02140000
##
## [1] "PM2.5MaxAvr"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 7e-05 (for x) and 6e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



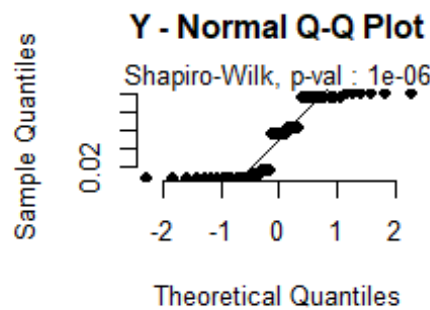
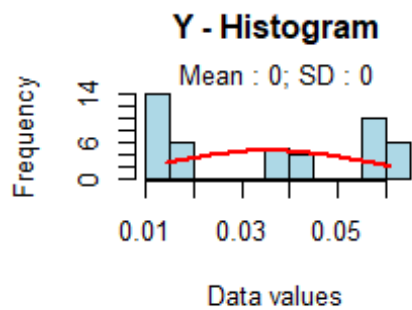
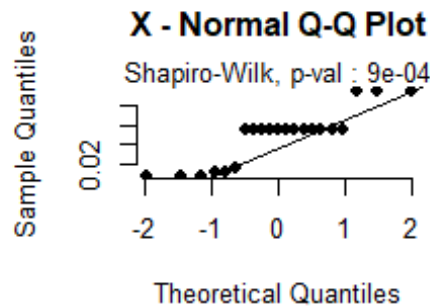
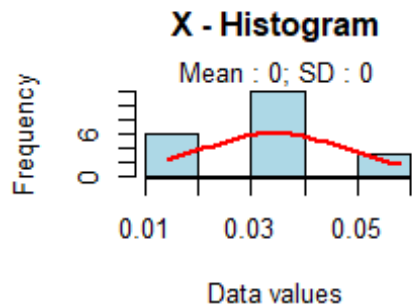
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.4325, df = 61.142, p-value = 0.001078
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.01153662 0.04373195
## sample estimates:
## mean of x mean of y
## 0.1630810 0.1354467
##
## [1] "PM2.5MaxMax"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 1e-05 (for x) and 1e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



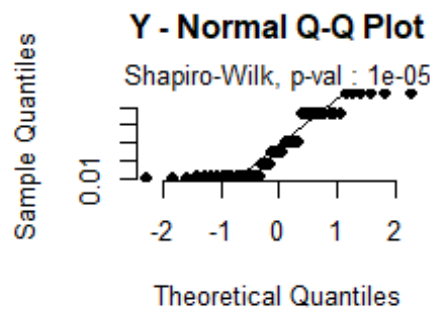
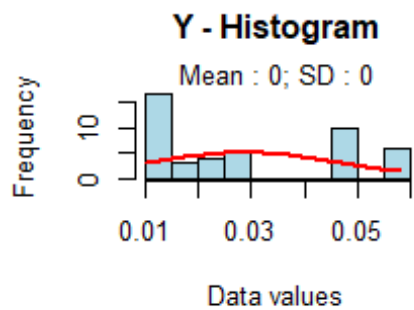
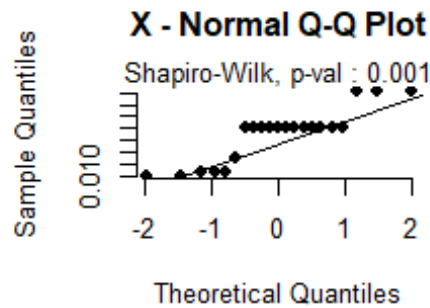
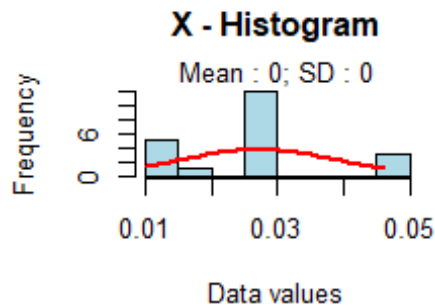
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 2.0007, df = 58.156, p-value = 0.0501
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.364467e-05 6.054063e-02
## sample estimates:
## mean of x mean of y
## 0.2546190 0.2243556
##
## [1] "TSPDT"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 9e-04 (for x) and 1e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```



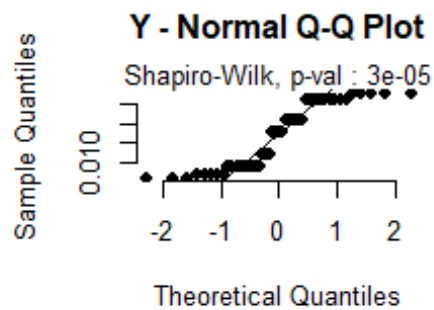
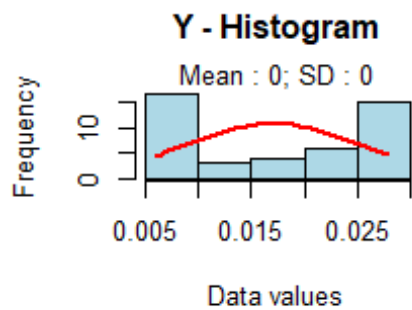
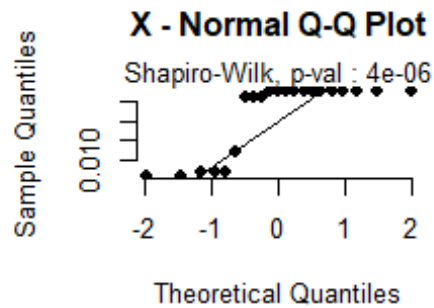
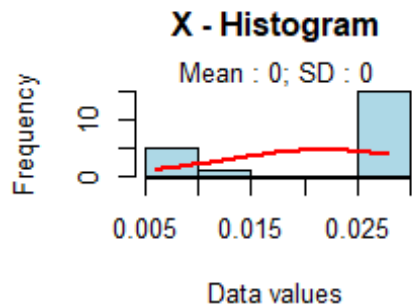
```
##
## Two Sample t-test
##
## data: x and y
## t = -0.25042, df = 64, p-value = 0.8031
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.010944062 0.008505966
## sample estimates:
## mean of x mean of y
## 0.03438095 0.03560000
##
## [1] "PM10DT"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.001 (for x) and 1e-05 (for y).
## Use a non parametric test like Wilcoxon test.
```



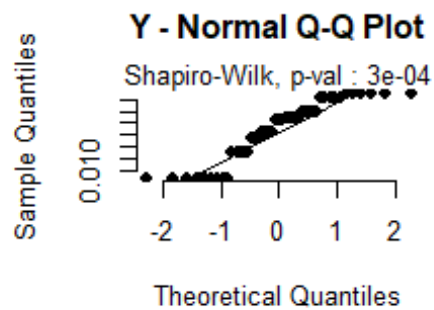
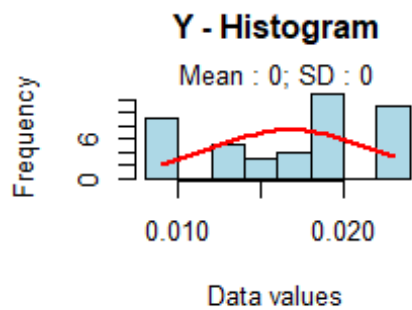
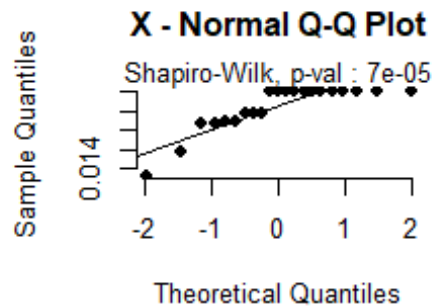
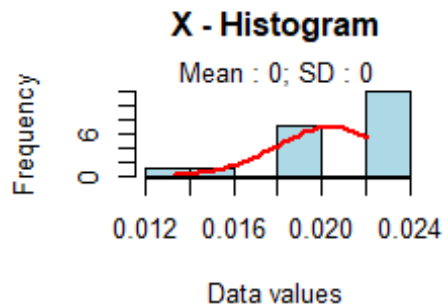
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = -0.43634, df = 57.466, p-value = 0.6642
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.008923783 0.005730133
## sample estimates:
## mean of x mean of y
## 0.02704762 0.02864444
##
## [1] "PM25DT"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 4e-06 (for x) and 3e-05 (for y).
## Use a non parametric test like Wilcoxon test.
```



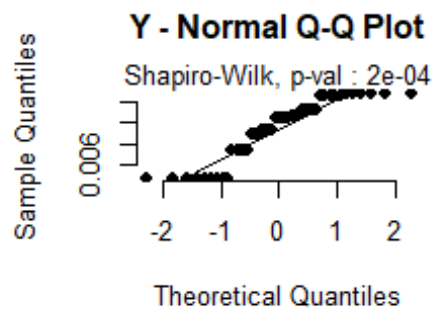
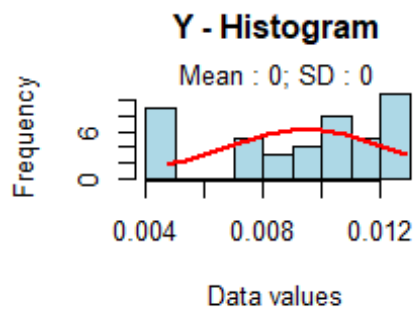
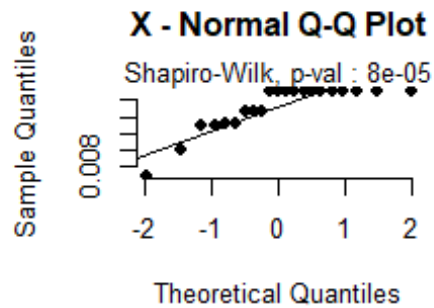
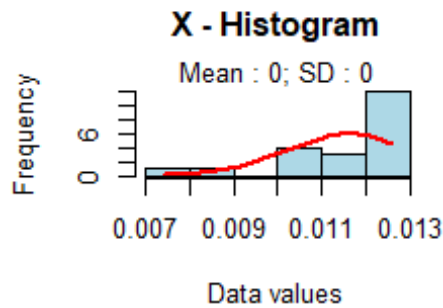
```
##
## Two Sample t-test
##
## data: x and y
## t = 2.0941, df = 64, p-value = 0.04021
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 0.0002205002 0.0093604522
## sample estimates:
## mean of x mean of y
## 0.02185714 0.01706667
##
## [1] "TBPLogMax"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 7e-05 (for x) and 3e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



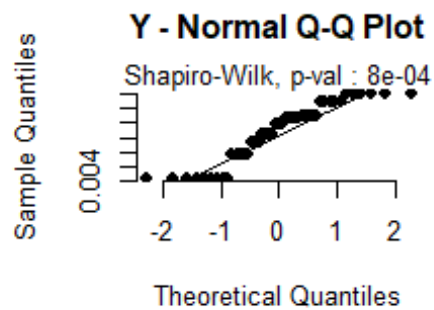
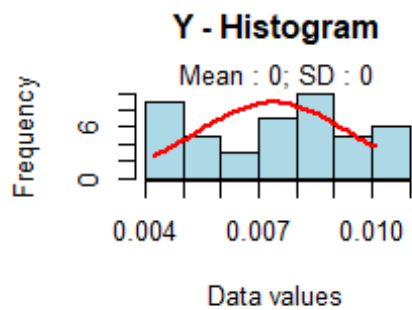
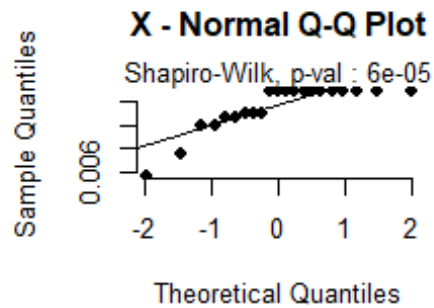
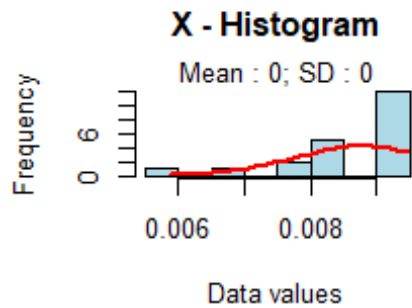
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.9356, df = 63.61, p-value = 0.0002082
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.001758815 0.005385987
## sample estimates:
##  mean of x mean of y
## 0.02038739 0.01681499
##
## [1] "PLogMax"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 8e-05 (for x) and 2e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



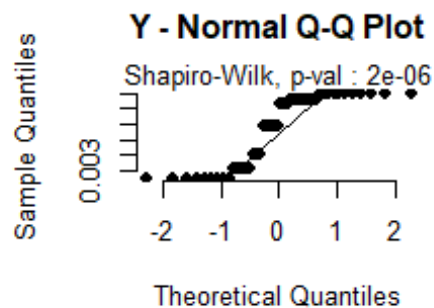
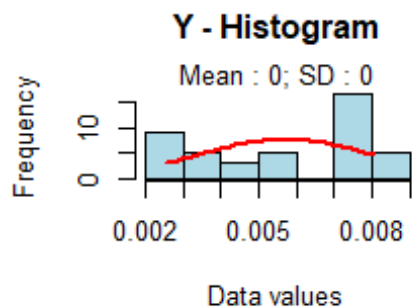
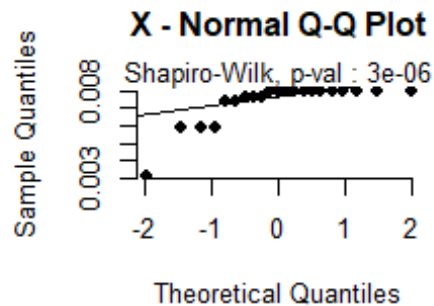
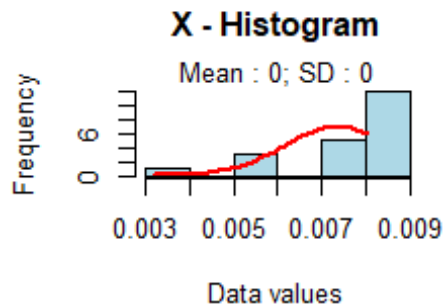
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 4.0625, df = 63.628, p-value = 0.0001357
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.001102165 0.003235352
## sample estimates:
##  mean of x   mean of y
## 0.011608775 0.009440017
##
## [1] "TBlogMax"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 6e-05 (for x) and 8e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```

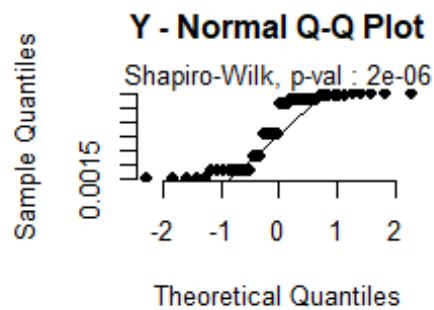
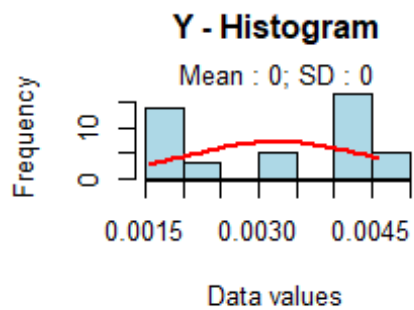
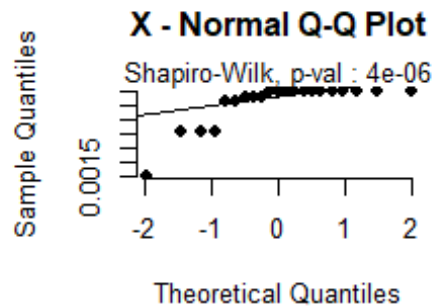
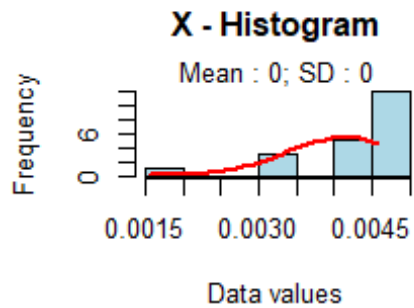
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.7435, df = 63.571, p-value = 0.0003927
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.0006544847 0.0021528019
## sample estimates:
##  mean of x   mean of y
## 0.008778618 0.007374975
##
## [1] "TBPLogAvr"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 3e-06 (for x) and 2e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```



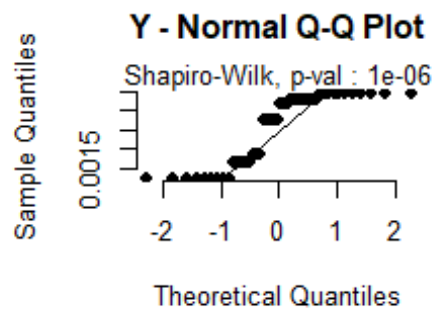
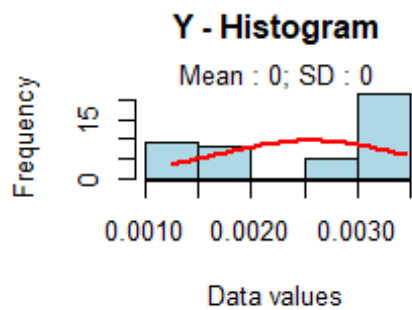
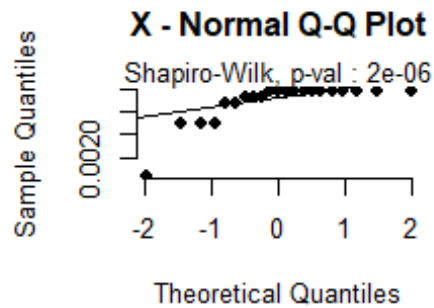
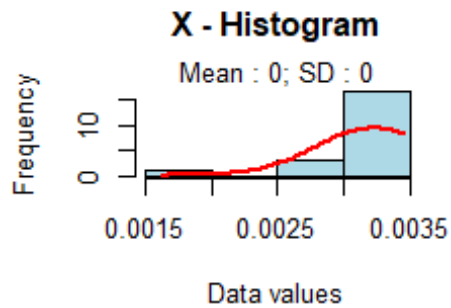
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.8152, df = 61.963, p-value = 0.0003164
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.0007906516 0.0025311054
## sample estimates:
##  mean of x   mean of y
## 0.007363400 0.005702521
##
## [1] "PLogAvr"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 4e-06 (for x) and 2e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```



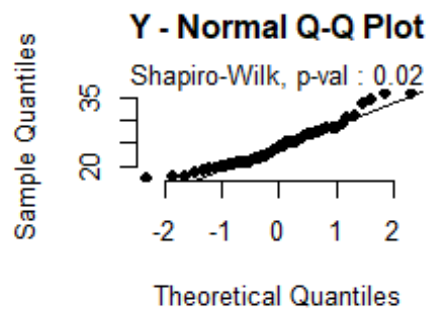
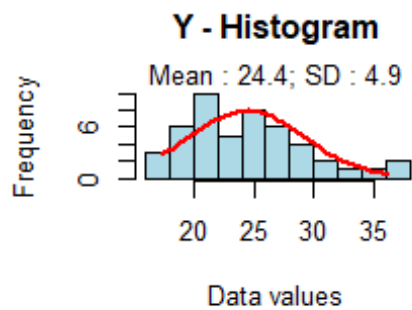
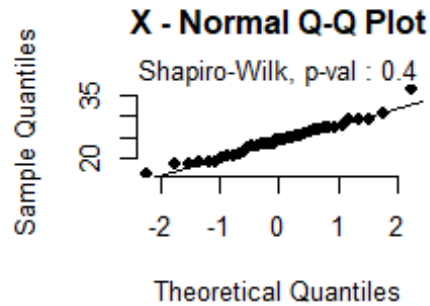
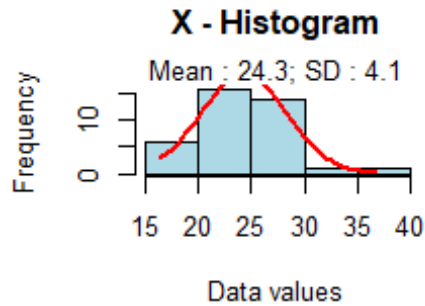
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.5281, df = 57.515, p-value = 0.0008303
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.0003848396 0.0013946075
## sample estimates:
##  mean of x   mean of y
## 0.004133337 0.003243613
##
## [1] "TLogAvr"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 2e-06 (for x) and 1e-06 (for y).
## Use a non parametric test like Wilcoxon test.
```



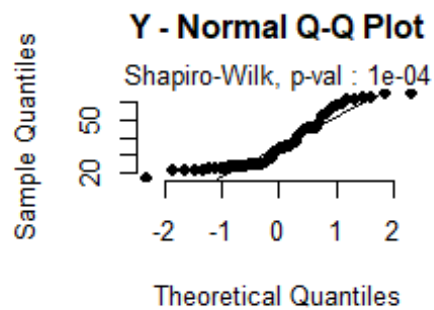
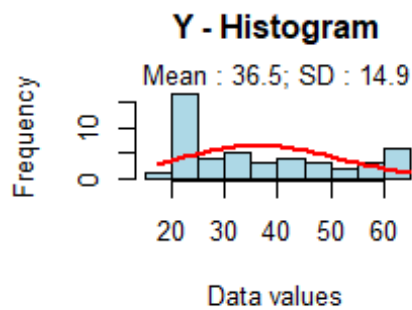
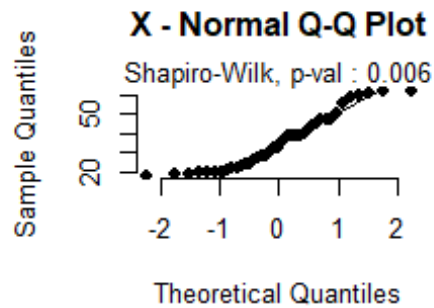
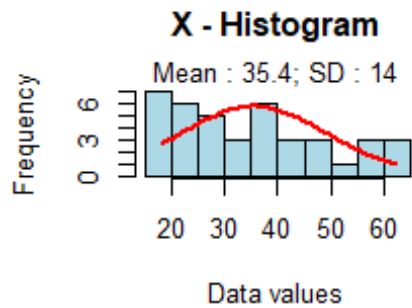
```
##
## Welch Two Sample t-test
##
## data: x and y
## t = 3.9352, df = 62.957, p-value = 0.0002102
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.0003273275 0.0010027864
## sample estimates:
##  mean of x   mean of y
## 0.003222938 0.002557881
##
## [1] "BMI"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.4 (for x) and 0.02 (for y).
## Use a non parametric test like Wilcoxon test.
```

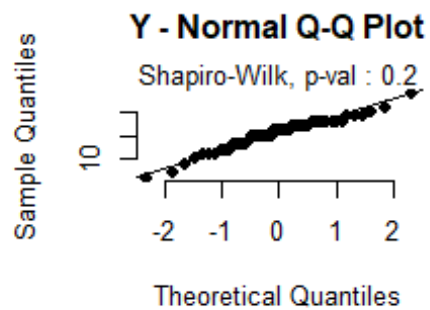
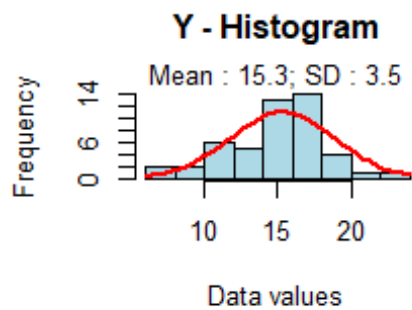
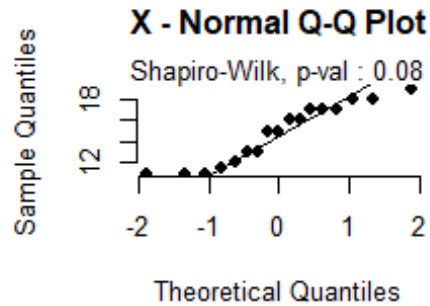
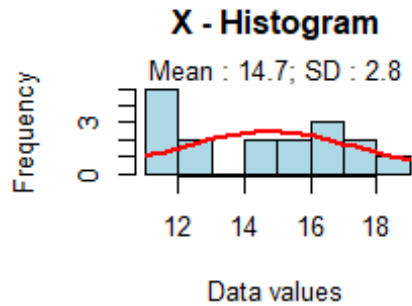


```
##
## Two Sample t-test
##
## data: x and y
## t = -0.1273, df = 84, p-value = 0.899
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -2.082447 1.831877
## sample estimates:
## mean of x mean of y
## 24.28263 24.40792
##
## [1] "Age"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.006 (for x) and 1e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```

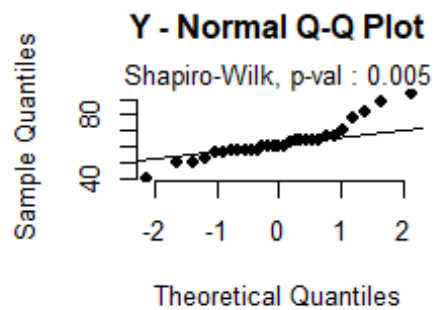
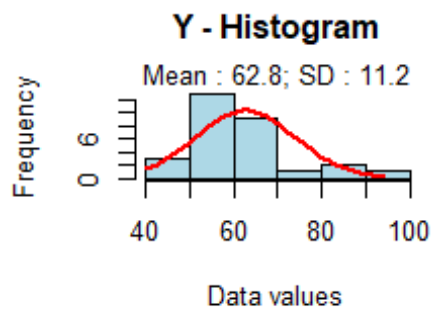
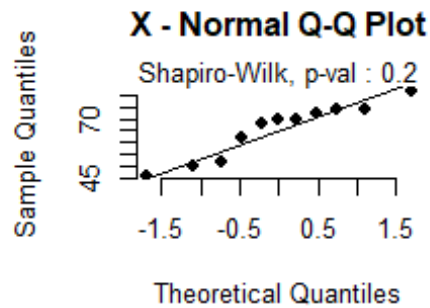
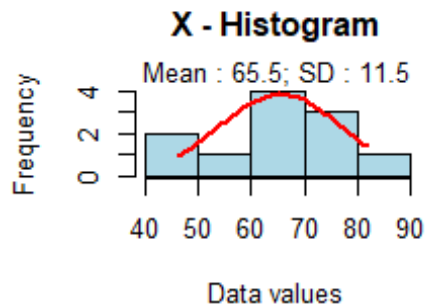


```
##
## Two Sample t-test
##
## data: x and y
## t = -0.33917, df = 86, p-value = 0.7353
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -7.211693 5.109501
## sample estimates:
## mean of x mean of y
## 35.44205 36.49315
##
## [1] "EducYrs"
```

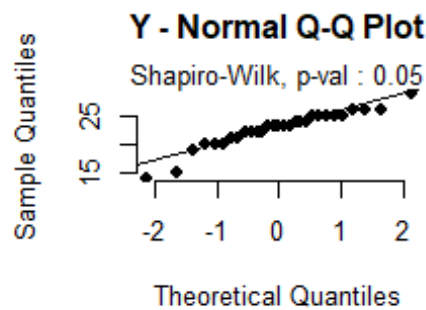
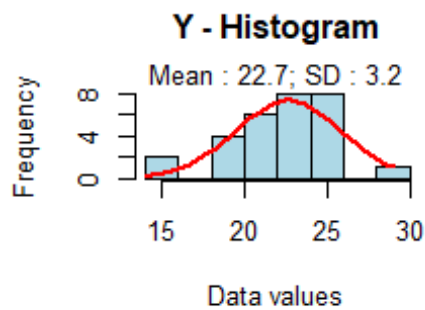
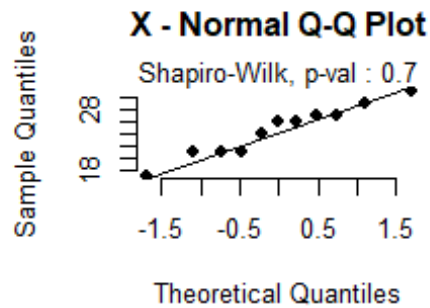
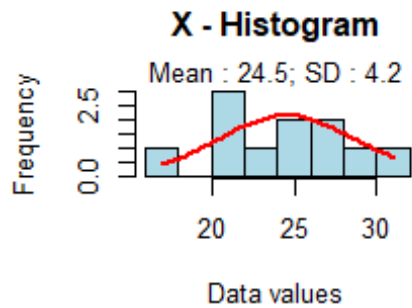


```
##
## Two Sample t-test
##
## data: x and y
## t = -0.63176, df = 63, p-value = 0.5298
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -2.489729 1.293650
## sample estimates:
## mean of x mean of y
## 14.73529 15.33333
##
## [1] "DL"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.2 (for x) and 0.005 (for y).
## Use a non parametric test like Wilcoxon test.
```

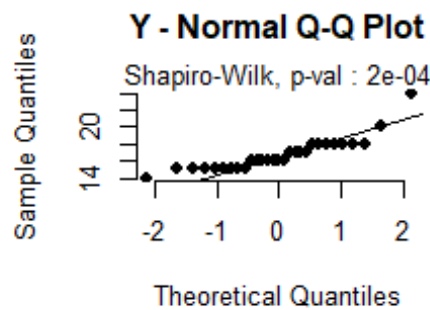
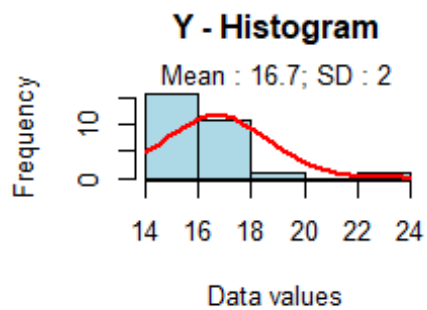
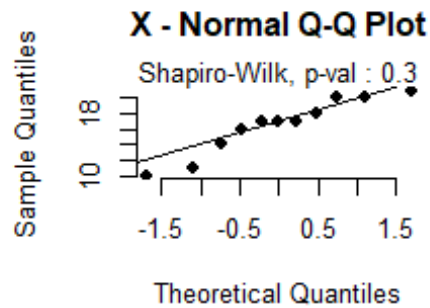
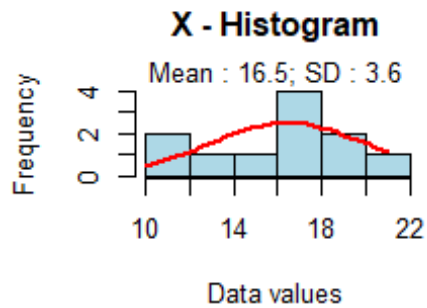


```
##
## Two Sample t-test
##
## data: x and y
## t = 0.67599, df = 38, p-value = 0.5031
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -5.377599 10.769449
## sample estimates:
## mean of x mean of y
## 65.45455 62.75862
##
## [1] "JD"
```

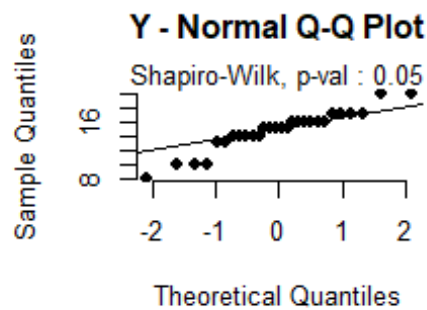
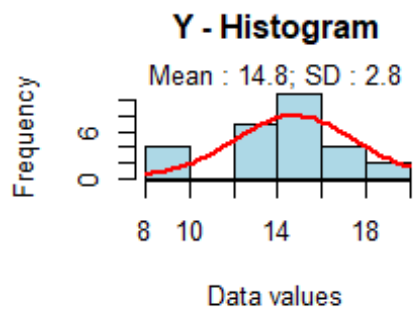
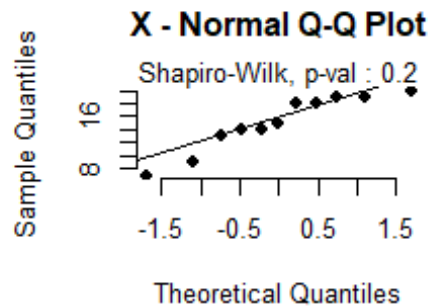
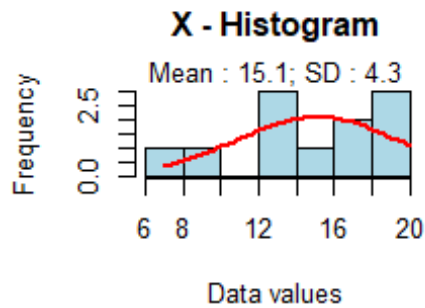



```
##
## Two Sample t-test
##
## data: x and y
## t = 1.5422, df = 38, p-value = 0.1313
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.5910034 4.3715676
## sample estimates:
## mean of x mean of y
## 24.54545 22.65517
##
## [1] "CS"

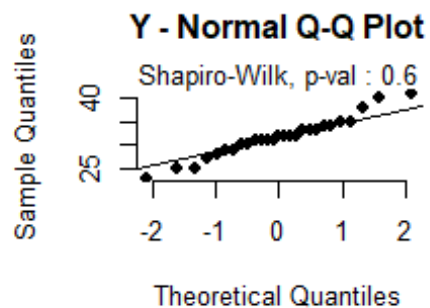
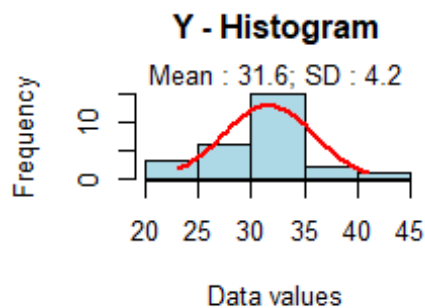
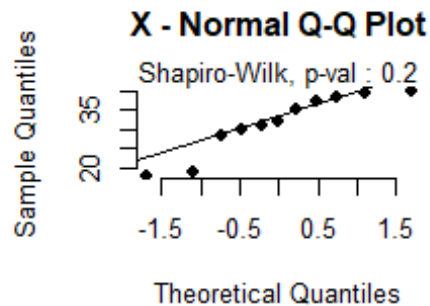
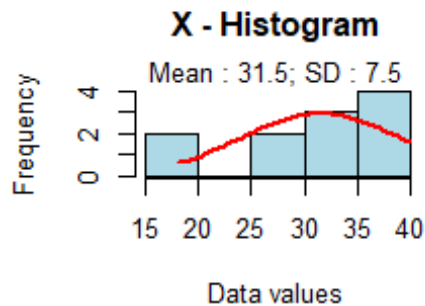
## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.3 (for x) and 2e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```



```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -0.20718, df = 12.433, p-value = 0.8392
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.698120  2.227901
## sample estimates:
## mean of x mean of y
##  16.45455  16.68966
##
## [1] "SS"
```

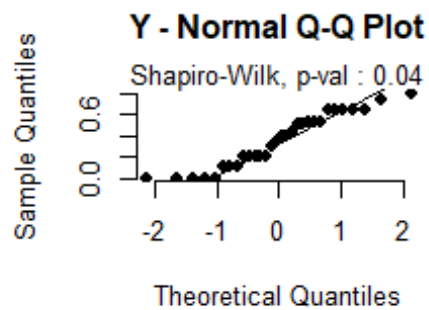
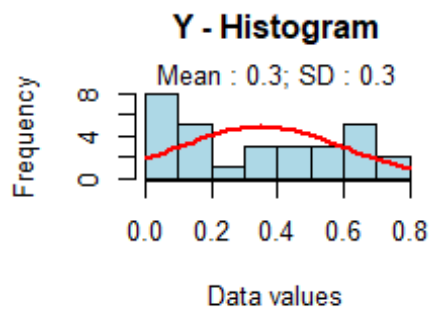
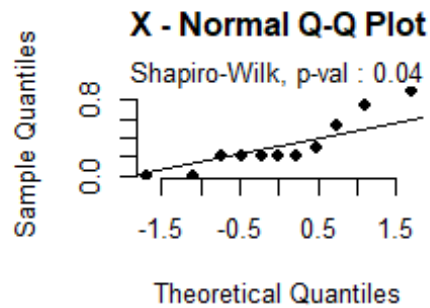
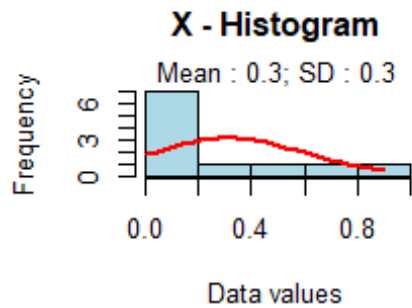


```
##
## Two Sample t-test
##
## data: x and y
## t = 0.29579, df = 37, p-value = 0.769
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.994385 2.676204
## sample estimates:
## mean of x mean of y
## 15.09091 14.75000
##
## [1] "SocSup"
```



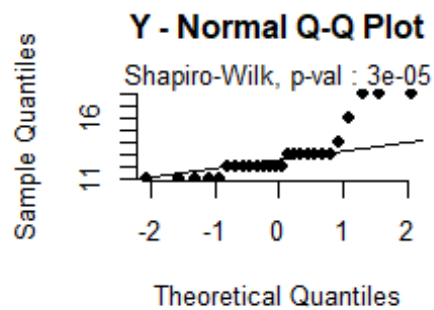
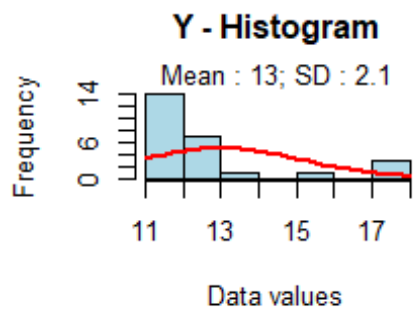
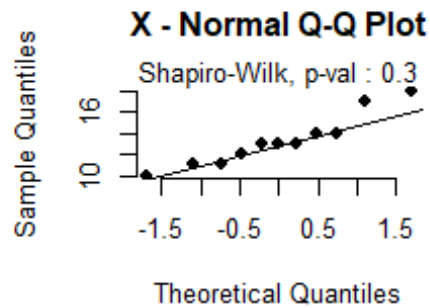
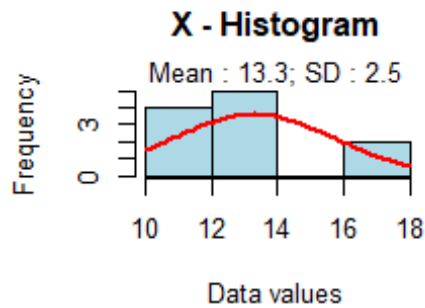
```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -0.034946, df = 12.605, p-value = 0.9727
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -5.304571  5.136221
## sample estimates:
## mean of x mean of y
##  31.54545  31.62963
##
## [1] "JobDis"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.04 (for x) and 0.04 (for y).
## Use a non parametric test like Wilcoxon test.
```

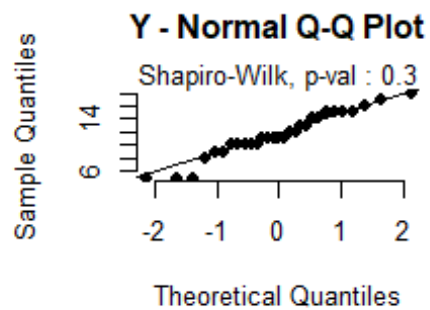
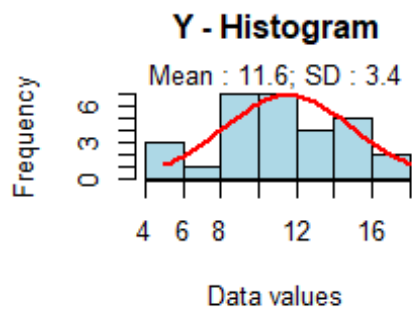
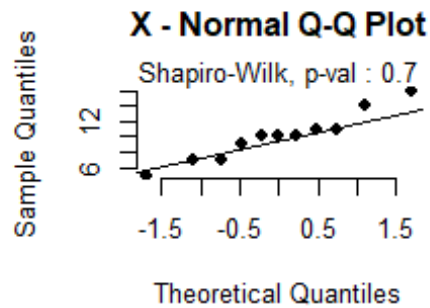
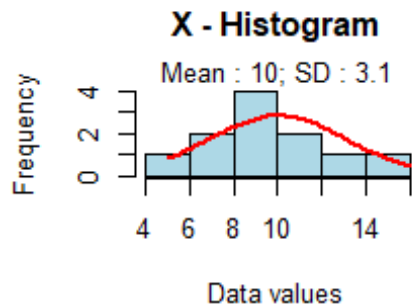


```
##
## Two Sample t-test
##
## data: x and y
## t = -0.36694, df = 39, p-value = 0.7156
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.2197086 0.1522338
## sample estimates:
## mean of x mean of y
## 0.3151515 0.3488889
##
## [1] "JI"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.3 (for x) and 3e-05 (for y).
## Use a non parametric test like Wilcoxon test.
```

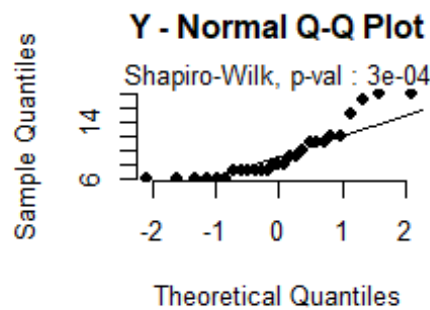
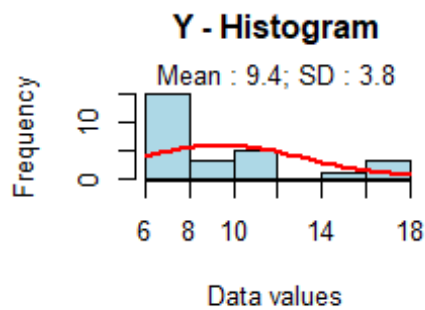
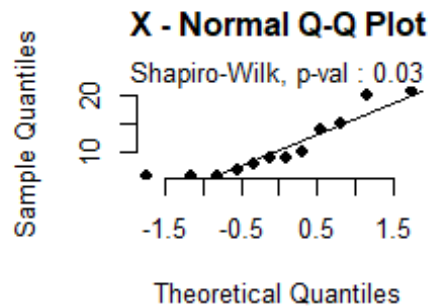
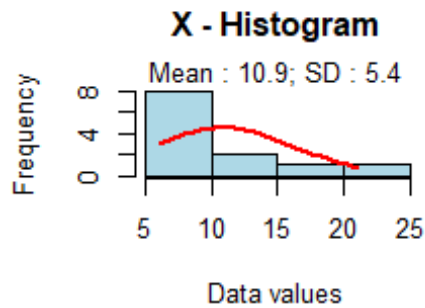


```
##
## Two Sample t-test
##
## data: x and y
## t = 0.33989, df = 35, p-value = 0.736
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.356210 1.901665
## sample estimates:
## mean of x mean of y
## 13.27273 13.00000
##
## [1] "PE"
```



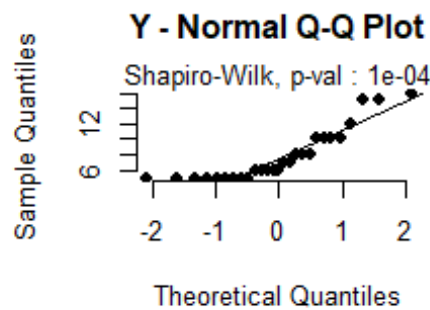
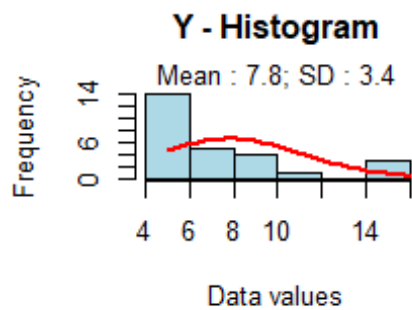
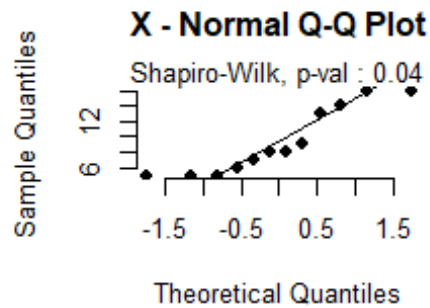
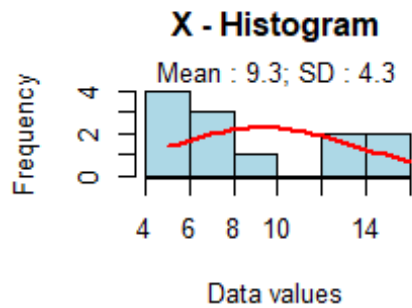
```
##
## Two Sample t-test
##
## data: x and y
## t = -1.3163, df = 38, p-value = 0.196
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -3.9382596 0.8348113
## sample estimates:
## mean of x mean of y
## 10.00000 11.55172
##
## [1] "ERI_Ef_Ph"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
normally distributed : Shapiro test p-value : 0.03 (for x) and 3e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```

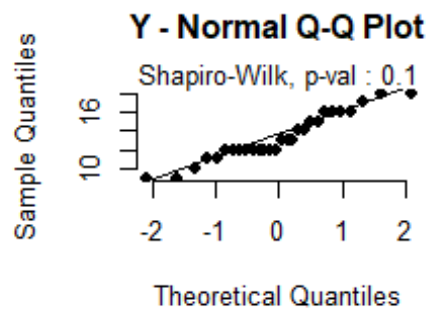
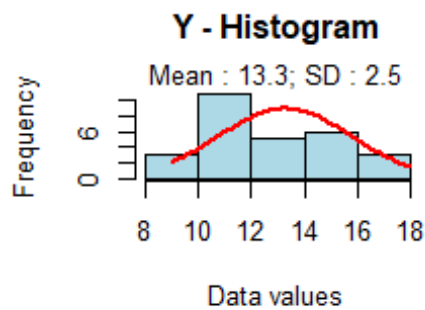
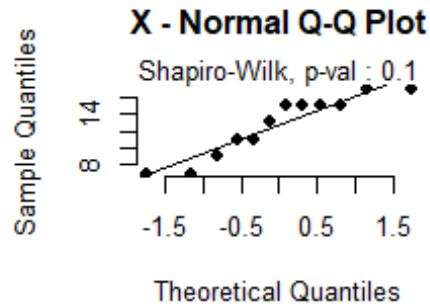
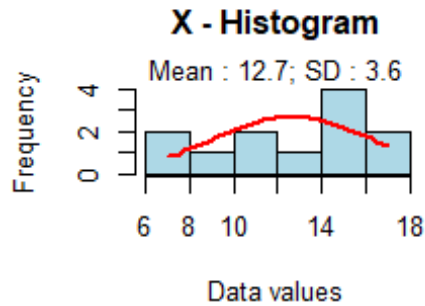


```
##
## Two Sample t-test
##
## data: x and y
## t = 0.98723, df = 37, p-value = 0.3299
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.549367 4.493811
## sample estimates:
## mean of x mean of y
## 10.916667 9.444444
##
## [1] "ERI_Ef_M"

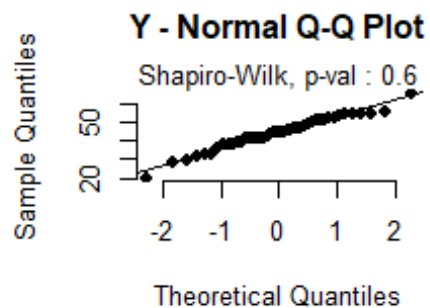
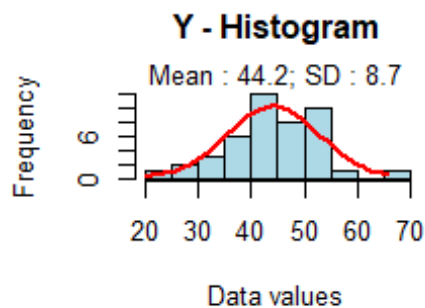
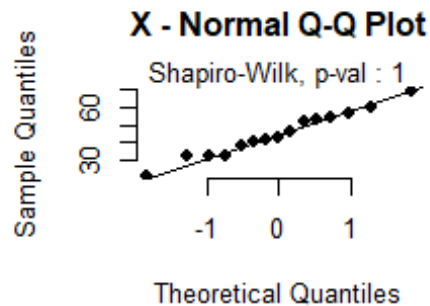
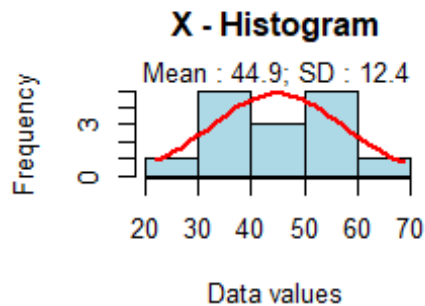
## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.04 (for x) and 1e-04 (for y).
## Use a non parametric test like Wilcoxon test.
```

```
##
## Two Sample t-test
##
## data: x and y
## t = 1.1945, df = 37, p-value = 0.2399
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.057236 4.094273
## sample estimates:
## mean of x mean of y
## 9.333333 7.814815
##
## [1] "Overcom"
```

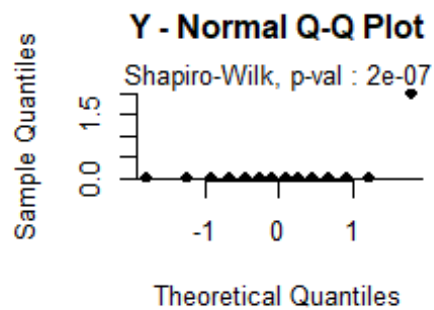
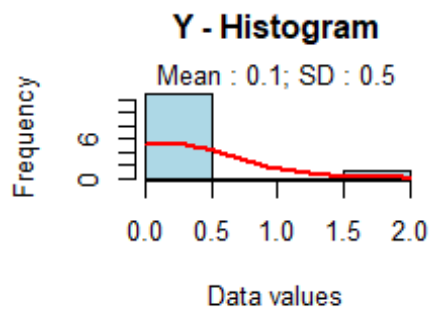
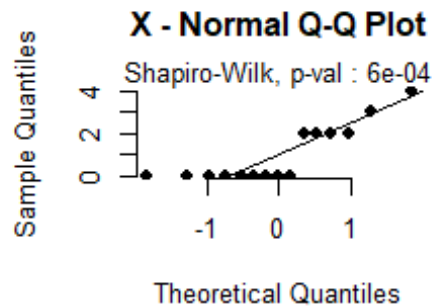
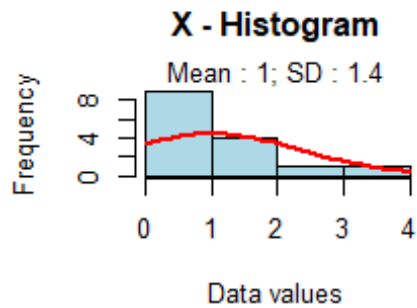


```
##
## Two Sample t-test
##
## data: x and y
## t = -0.62778, df = 38, p-value = 0.5339
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -2.615280 1.377185
## sample estimates:
## mean of x mean of y
## 12.66667 13.28571
##
## [1] "SAnx"
```

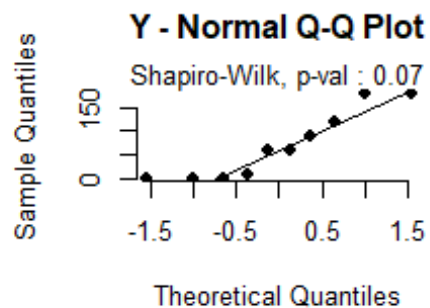
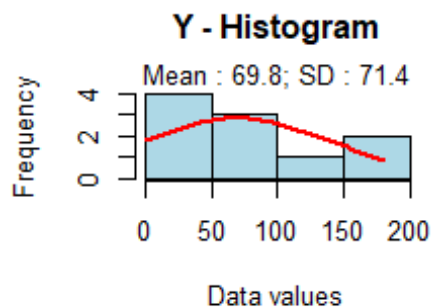
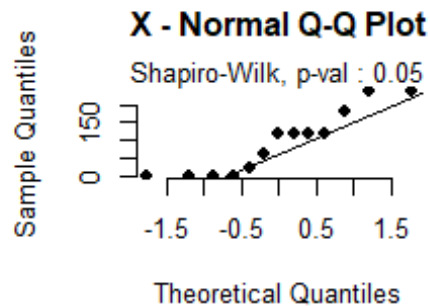
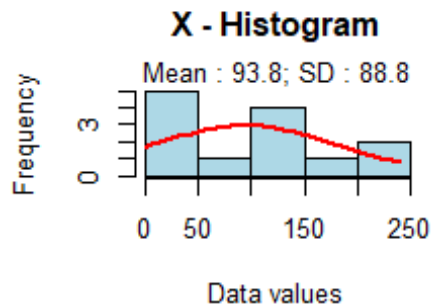


```
##
## Two Sample t-test
##
## data: x and y
## t = 0.26525, df = 57, p-value = 0.7918
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -5.070889 6.619374
## sample estimates:
## mean of x mean of y
## 44.93333 44.15909
##
## [1] "PasSMK"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 6e-04 (for x) and 2e-07 (for y).
## Use a non parametric test like Wilcoxon test.
```

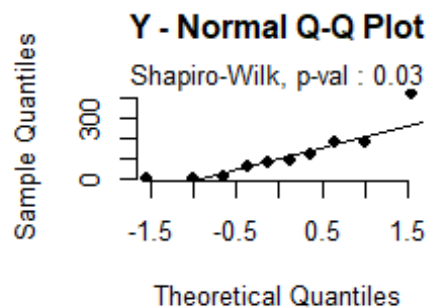
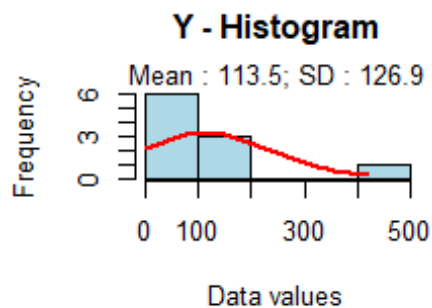
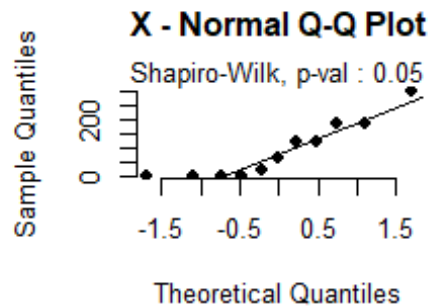
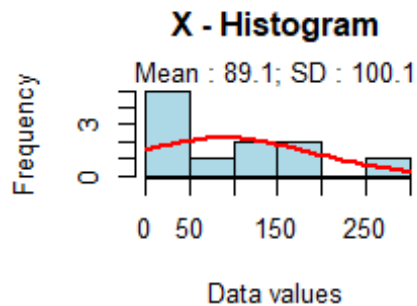


```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = 2.2571, df = 18.456, p-value = 0.03634
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.06070651 1.65357920
## sample estimates:
## mean of x mean of y
## 1.0000000 0.1428571
##
## [1] "PA_RigMin"
```



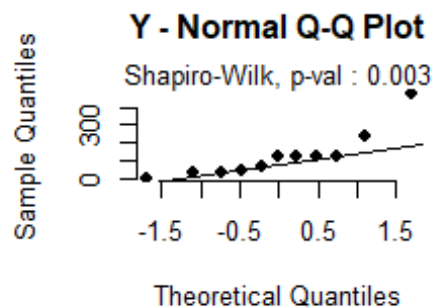
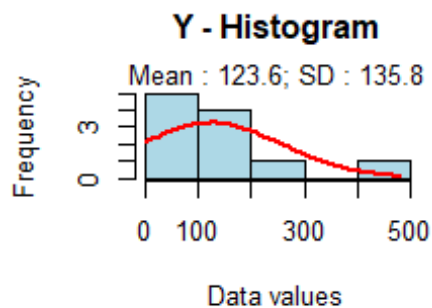
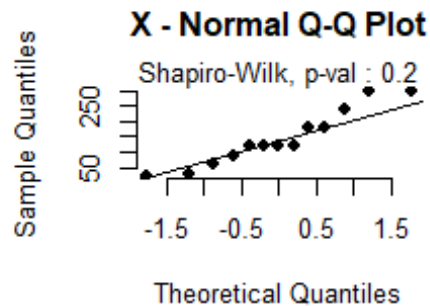
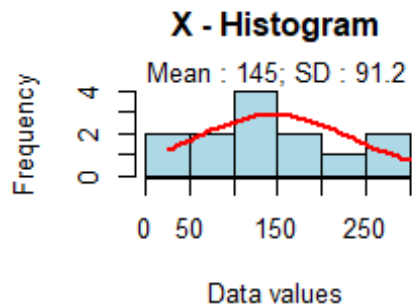
```
##
## Two Sample t-test
##
## data: x and y
## t = 0.69879, df = 21, p-value = 0.4924
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -47.51589 95.60819
## sample estimates:
## mean of x mean of y
## 93.84615 69.80000
##
## [1] "PA_ModMin"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.05 (for x) and 0.03 (for y).
## Use a non parametric test like Wilcoxon test.
```



```
##
## Two Sample t-test
##
## data: x and y
## t = -0.49177, df = 19, p-value = 0.6285
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -128.29703 79.47885
## sample estimates:
## mean of x mean of y
## 89.09091 113.50000
##
## [1] "Walk_Min"

## Warning in rquery.t.test(exp_n.BA12[[i]], exp_n.BA02[[i]]): x or y is not
## normally distributed : Shapiro test p-value : 0.2 (for x) and 0.003 (for y).
## Use a non parametric test like Wilcoxon test.
```



```
##
## Two Sample t-test
##
## data: x and y
## t = 0.45869, df = 22, p-value = 0.651
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -75.22778 117.95505
## sample estimates:
## mean of x mean of y
## 145.0000 123.6364

#wilcox-mecm
for (i in 1:num2) {
  print(names(exp_n2[i]))
  print(wilcox.test(exp_n.BA12[[i]], exp_n.BA02[[i]]), alternative = "g")
}

## [1] "TSPAvr"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
## compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 661.5, p-value = 0.008594
## alternative hypothesis: true location shift is not equal to 0
```

```

##
## [1] "TSPMaxAvr"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 635.5, p-value = 0.02349
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TSPMaxMax"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 557.5, p-value = 0.2371
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM10Avr"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 733.5, p-value = 0.0002818
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM10MaxAvr"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 630.5, p-value = 0.02812
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM10MaxMax"

```



```

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 638, p-value = 0.02122
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM2.5Avr"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 706.5, p-value = 0.001111
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM2.5MaxAvr"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 627.5, p-value = 0.03125
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM2.5MaxMax"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 617.5, p-value = 0.04353
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TSPDT"

```

```

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 442.5, p-value = 0.678
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM10DT"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 471.5, p-value = 0.9944
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PM25DT"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 653, p-value = 0.01185
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TBPLogMax"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 657.5, p-value = 0.01011
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PLogMax"

```

```

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 661.5, p-value = 0.008594
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TBLogMax"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 657.5, p-value = 0.01011
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TBPLogAvr"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 702, p-value = 0.0006504
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PLogAvr"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 693, p-value = 0.001032
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "TBLogAvr"

```

```

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 702, p-value = 0.0006504
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "BMI"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 926.5, p-value = 0.9031
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "Age"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 907, p-value = 0.66
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "EducYrs"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 355.5, p-value = 0.4347
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "DL"

```

```

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 197.5, p-value = 0.2543
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "JD"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 208, p-value = 0.144
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "CS"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 173.5, p-value = 0.6781
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "SS"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 167.5, p-value = 0.6827
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "SocSup"

```

```

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 160.5, p-value = 0.7104
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "JobDis"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 151, p-value = 0.6878
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "JI"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 157.5, p-value = 0.6336
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PE"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 112.5, p-value = 0.1556
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "ERI_Ef_Ph"

```

```

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 181.5, p-value = 0.5591
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "ERI_Ef_M"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 195, p-value = 0.3137
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "Overcom"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 155.5, p-value = 0.7207
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "SANx"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 341.5, p-value = 0.8479
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PasSMK"

```

```

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 140.5, p-value = 0.04102
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PA_RigMin"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 74, p-value = 0.5902
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "PA_ModMin"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 48.5, p-value = 0.6675
## alternative hypothesis: true location shift is not equal to 0
##
## [1] "Walk_Min"

## Warning in wilcox.test.default(exp_n.BA12[[i]], exp_n.BA02[[i]]): cannot
compute
## exact p-value with ties

##
## Wilcoxon rank sum test with continuity correction
##
## data: exp_n.BA12[[i]] and exp_n.BA02[[i]]
## W = 88, p-value = 0.3441
## alternative hypothesis: true location shift is not equal to 0

#j45.1
#сравнение BA0 и BA1 по переменным
#критерий Wilcox
#"BMI", "Age", "DL", "ERI_Ef_Ph", "ERI_Ef_M", "Overcom"

```



```
#for (i in 1:num) {
#  print(names(spl.exp_n1)[[i]])
#  print(lapply(spl.exp_n1[[i]], f.numstat))
#}
```

```
#j45.0
#сравнение BA0 и BA1 по переменным
#критерий Wilcox
#"PasSMK"
```

```
for (i in 1:num2) {
  print(names(spl.exp_n2)[[i]])
  print(lapply(spl.exp_n2[[i]], f.numstat))
}
```

```
## [1] "TSPAvr"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.1070  0.1140  0.1450  0.1374  0.1590  0.1660         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.114  0.150  0.164  0.156  0.164  0.164        19
##
## [1] "TSPMaxAvr"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.3623  0.4475  0.5352  0.5163  0.5952  0.6332         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.4998  0.5518  0.5952  0.5753  0.5952  0.6332        19
##
## [1] "TSPMaxMax"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.4950  0.6720  0.7200  0.6824  0.7480  0.7690         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.6720  0.7200  0.7340  0.7299  0.7340  0.7690        19
##
## [1] "PM10Avr"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.04300 0.04800 0.07900 0.06951 0.08500 0.09400         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.04800 0.08100 0.09400 0.08662 0.09400 0.09400        19
##
```

```

## [1] "PM10MaxAvr"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.1715  0.2025  0.2620  0.2706  0.3485  0.3797         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.2025  0.2773  0.3485  0.3223  0.3485  0.3635        19
##
## [1] "PM10MaxMax"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.261   0.291   0.420   0.384   0.440   0.453         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.291   0.430   0.440   0.431   0.440   0.453        19
##
## [1] "PM2.5Avr"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.0100  0.0120  0.0240  0.0214  0.0290  0.0320         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.01200 0.02500 0.03100 0.02838 0.03100 0.03200        19
##
## [1] "PM2.5MaxAvr"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.0583  0.1062  0.1417  0.1354  0.1720  0.1985         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.1062  0.1555  0.1720  0.1631  0.1720  0.1850        19
##
## [1] "PM2.5MaxMax"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.1270  0.1530  0.2300  0.2244  0.2530  0.3790         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.1530  0.2400  0.2530  0.2546  0.2530  0.3790        19
##
## [1] "TSPDT"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.0140  0.0140  0.0380  0.0356  0.0580  0.0610         3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's

```

```

## 0.01400 0.01800 0.03800 0.03438 0.03800 0.05800      19
##
## [1] "PM10DT"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.01000 0.01100 0.02400 0.02864 0.04600 0.05800      3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.01000 0.01700 0.03000 0.02705 0.03000 0.04600      19
##
## [1] "PM25DT"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.00600 0.00900 0.01800 0.01707 0.02600 0.02800      3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.00600 0.01200 0.02800 0.02186 0.02800 0.02800      19
##
## [1] "TBPLogMax"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.008971 0.013276 0.018618 0.016815 0.019761 0.022998      3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.01328 0.01878 0.02210 0.02039 0.02210 0.02210      19
##
## [1] "PLogMax"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.004758 0.007416 0.010486 0.009440 0.011298 0.012901      3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.007416 0.010625 0.012611 0.011609 0.012611 0.012611      19
##
## [1] "TBLogMax"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.004213 0.005860 0.007993 0.007375 0.008463 0.010097      3
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.005860 0.008291 0.009484 0.008779 0.009484 0.009484      19
##
## [1] "TBPLogAvr"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.002546 0.003168 0.006599 0.005703 0.007674 0.008038      4
##

```

```

## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.003168 0.007346 0.008038 0.007363 0.008038 0.008038      19
##
## [1] "PLogAvr"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.001545 0.001812 0.003620 0.003244 0.004333 0.004569       4
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.001545 0.004152 0.004569 0.004133 0.004569 0.004569      19
##
## [1] "TBlogAvr"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.001243 0.001615 0.002968 0.002558 0.003339 0.003462       4
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 0.001615 0.003182 0.003462 0.003223 0.003462 0.003462      19
##
## [1] "BMI"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
## 17.30    20.70    23.84    24.41    27.38    36.20
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 16.41    21.37    24.22    24.28    26.82    36.74       2
##
## [1] "Age"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
## 17.37    23.48    33.08    36.49    46.09    64.89
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
## 17.97    22.42    33.54    35.44    45.30    62.33
##
## [1] "EducYrs"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
## 6.00     13.00    16.00    15.33    18.00    24.00
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
## 11.00    12.00    15.00    14.74    17.00    19.00      23
##
## [1] "DL"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's

```

```

## 40.00 58.00 60.00 62.76 64.00 94.00 19
##
## $`1`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 46.00 57.00 70.00 65.45 73.00 82.00 29
##
## [1] "JD"
## $`0`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 14.00 21.00 23.00 22.66 25.00 29.00 19
##
## $`1`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 17.00 21.00 26.00 24.55 27.00 31.00 29
##
## [1] "CS"
## $`0`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 14.00 15.00 16.00 16.69 18.00 24.00 19
##
## $`1`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 10.00 15.00 17.00 16.45 19.00 21.00 29
##
## [1] "SS"
## $`0`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 8.00 14.00 15.00 14.75 16.00 20.00 20
##
## $`1`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 7.00 13.50 15.00 15.09 18.50 20.00 29
##
## [1] "SocSup"
## $`0`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 23.00 29.50 32.00 31.63 33.50 41.00 21
##
## $`1`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 18.00 29.00 32.00 31.55 37.50 40.00 29
##
## [1] "JobDis"
## $`0`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 0.0000 0.1250 0.3667 0.3489 0.5333 0.8000 18
##
## $`1`
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 0.0000 0.2000 0.2000 0.3152 0.4167 0.9000 29
##
## [1] "JI"

```

```

## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##      11      12      12       13      13      18       22
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##     10.00  11.50   13.00   13.27  14.00   18.00       29
##
## [1] "PE"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##      5.00   10.00   11.00   11.55  14.00   18.00       19
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##       5        8       10       10      11      16       29
##
## [1] "ERI_Ef_Ph"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##     6.000   7.000   8.000   9.444  11.000   18.000       21
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##     6.00   6.75   9.00   10.92  14.25   21.00       28
##
## [1] "ERI_Ef_M"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##     5.000   5.000   6.000   7.815  10.000   16.000       21
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##     5.000   5.750   8.000   9.333  13.250   16.000       28
##
## [1] "Overcom"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##      9.00   12.00   12.50   13.29  15.25   18.00       20
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##      7.00   10.50   14.00   12.67  15.00   17.00       28
##
## [1] "SAnx"
## $`0`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##     20.00   39.00   45.00   44.16  51.00   66.00        4
##
## $`1`
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.      NA's
##     22.00   35.50   43.00   44.93  53.50   69.00       25

```

```
##
## [1] "PasSMK"
## $`0`
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##  0.0000  0.0000  0.0000  0.1429  0.0000  2.0000    34
##
## $`1`
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##         0         0         0         1         2         4    25
##
## [1] "PA_RigMin"
## $`0`
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      0.0      2.0     60.0    69.8   112.5   180.0    38
##
## $`1`
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      0.00     0.00   120.00   93.85   120.00   240.00    27
##
## [1] "PA_ModMin"
## $`0`
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      0.00    18.75    85.00   113.50   165.00   420.00    38
##
## $`1`
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      0.00     0.00    60.00    89.09   150.00   300.00    29
##
## [1] "Walk_Min"
## $`0`
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      0.0     35.0    120.0   123.6   120.0   480.0    37
##
## $`1`
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   NA's
##      25      90     120     145     180     300    27

chisq.test(ba1$BA, ba1$Atopia)

## Warning in chisq.test(ba1$BA, ba1$Atopia): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba1$BA and ba1$Atopia
## X-squared = 1.5783, df = 1, p-value = 0.209

chisq.test(ba1$BA, ba1$Atopia, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
```

```

##
## data:  ba1$BA and ba1$Atopia
## X-squared = 3.4017, df = NA, p-value = 0.1034

chisq.test(ba1$BA, ba1$Heredity)

##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba1$BA and ba1$Heredity
## X-squared = 10.382, df = 1, p-value = 0.001272

chisq.test(ba1$BA, ba1$Heredity, simulate.p.value = TRUE)

##
##  Pearson's Chi-squared test with simulated p-value (based on 2000
##  replicates)
##
## data:  ba1$BA and ba1$Heredity
## X-squared = 12.154, df = NA, p-value = 0.001499

chisq.test(ba1$BA, ba1$Sex)

##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba1$BA and ba1$Sex
## X-squared = 3.3254, df = 1, p-value = 0.06822

chisq.test(ba1$BA, ba1$Family)

## Warning in chisq.test(ba1$BA, ba1$Family): Chi-squared approximation may be
## incorrect

##
##  Pearson's Chi-squared test
##
## data:  ba1$BA and ba1$Family
## X-squared = 11.367, df = 5, p-value = 0.04456

chisq.test(ba1$BA, ba1$Family, simulate.p.value = TRUE)

##
##  Pearson's Chi-squared test with simulated p-value (based on 2000
##  replicates)
##
## data:  ba1$BA and ba1$Family
## X-squared = 11.367, df = NA, p-value = 0.03848

chisq.test(ba1$BA, ba1$FamType)

## Warning in chisq.test(ba1$BA, ba1$FamType): Chi-squared approximation may be
## incorrect

```



```

##
## Pearson's Chi-squared test
##
## data:  ba1$BA and ba1$FamType
## X-squared = 4.8939, df = 4, p-value = 0.2984

chisq.test(ba1$BA, ba1$FamType, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data:  ba1$BA and ba1$FamType
## X-squared = 4.8939, df = NA, p-value = 0.3068

chisq.test(ba1$BA, ba1$Child)

## Warning in chisq.test(ba1$BA, ba1$Child): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data:  ba1$BA and ba1$Child
## X-squared = 9.5945, df = 5, p-value = 0.08757

chisq.test(ba1$BA, ba1$Child, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data:  ba1$BA and ba1$Child
## X-squared = 9.5945, df = NA, p-value = 0.06697

chisq.test(ba1$BA, ba1$Child_8)

## Warning in chisq.test(ba1$BA, ba1$Child_8): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba1$BA and ba1$Child_8
## X-squared = 6.8967e-31, df = 1, p-value = 1

chisq.test(ba1$BA, ba1$Child_8, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data:  ba1$BA and ba1$Child_8
## X-squared = 0.02374, df = NA, p-value = 1

```

```

chisq.test(ba1$BA, ba1$Hobby)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$Hobby
## X-squared = 1.8274, df = 1, p-value = 0.1764

chisq.test(ba1$BA, ba1$Educ)

## Warning in chisq.test(ba1$BA, ba1$Educ): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba1$BA and ba1$Educ
## X-squared = 8.0698, df = 4, p-value = 0.08905

chisq.test(ba1$BA, ba1$Educ, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$Educ
## X-squared = 8.0698, df = NA, p-value = 0.08396

chisq.test(ba1$BA, ba1$ScaleS)

## Warning in chisq.test(ba1$BA, ba1$ScaleS): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba1$BA and ba1$ScaleS
## X-squared = 10.877, df = 6, p-value = 0.09224

chisq.test(ba1$BA, ba1$ScaleS, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$ScaleS
## X-squared = 10.877, df = NA, p-value = 0.08896

chisq.test(ba1$BA, ba1$ScaleR)

## Warning in chisq.test(ba1$BA, ba1$ScaleR): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test

```

```

##
## data:  ba1$BA and ba1$ScaleR
## X-squared = 6.2136, df = 8, p-value = 0.6233

chisq.test(ba1$BA, ba1$ScaleR, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data:  ba1$BA and ba1$ScaleR
## X-squared = 6.2136, df = NA, p-value = 0.6552

chisq.test(ba1$BA, ba1$ISL_cd)

## Warning in chisq.test(ba1$BA, ba1$ISL_cd): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data:  ba1$BA and ba1$ISL_cd
## X-squared = 1.7342, df = 3, p-value = 0.6294

chisq.test(ba1$BA, ba1$ISL_cd, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data:  ba1$BA and ba1$ISL_cd
## X-squared = 1.7342, df = NA, p-value = 0.6572

chisq.test(ba1$BA, ba1$ProfCateg)

## Warning in chisq.test(ba1$BA, ba1$ProfCateg): Chi-squared approximation may
## be
## incorrect

##
## Pearson's Chi-squared test
##
## data:  ba1$BA and ba1$ProfCateg
## X-squared = 8.8183, df = 3, p-value = 0.03181

chisq.test(ba1$BA, ba1$ProfCateg, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data:  ba1$BA and ba1$ProfCateg
## X-squared = 8.8183, df = NA, p-value = 0.03048

chisq.test(ba1$BA, ba1$DustCurrent)

```

```

## Warning in chisq.test(ba1$BA, ba1$DustCurrent): Chi-squared approximation may
be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$DustCurrent
## X-squared = 2.246, df = 1, p-value = 0.134

chisq.test(ba1$BA, ba1$DustCurrent, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$DustCurrent
## X-squared = 3.3907, df = NA, p-value = 0.09445

chisq.test(ba1$BA, ba1$Dust)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$Dust
## X-squared = 0, df = 1, p-value = 1

chisq.test(ba1$BA, ba1$ChemCurrent)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$ChemCurrent
## X-squared = 0.0047827, df = 1, p-value = 0.9449

chisq.test(ba1$BA, ba1$Chem)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$Chem
## X-squared = 0.1055, df = 1, p-value = 0.7453

chisq.test(ba1$BA, ba1$BioCurrent)

## Warning in chisq.test(ba1$BA, ba1$BioCurrent): Chi-squared approximation may
be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$BioCurrent
## X-squared = 4.672, df = 1, p-value = 0.03066

```

```

chisq.test(ba1$BA, ba1$BioCurrent, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$BioCurrent
## X-squared = 7.9342, df = NA, p-value = 0.01949

chisq.test(ba1$BA, ba1$Bio)

## Warning in chisq.test(ba1$BA, ba1$Bio): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$Bio
## X-squared = 1.98e-31, df = 1, p-value = 1

chisq.test(ba1$BA, ba1$Bio, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$Bio
## X-squared = 0.041071, df = NA, p-value = 1

chisq.test(ba1$BA, ba1$Cold)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$Cold
## X-squared = 0.52248, df = 1, p-value = 0.4698

chisq.test(ba1$BA, ba1$Heat)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$Heat
## X-squared = 0.69308, df = 1, p-value = 0.4051

chisq.test(ba1$BA, ba1$PhysLoad)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$PhysLoad
## X-squared = 0.048526, df = 1, p-value = 0.8256

chisq.test(ba1$BA, ba1$SMK)

```

```

## Warning in chisq.test(ba1$BA, ba1$SMK): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$SMK
## X-squared = 2.6323e-31, df = 1, p-value = 1

chisq.test(ba1$BA, ba1$SMK, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$SMK
## X-squared = 0.30139, df = NA, p-value = 1

chisq.test(ba1$BA, ba1$FruVegs)

## Warning in chisq.test(ba1$BA, ba1$FruVegs): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$FruVegs
## X-squared = 0.45812, df = 1, p-value = 0.4985

chisq.test(ba1$BA, ba1$FruVegs, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$FruVegs
## X-squared = 1.1508, df = NA, p-value = 0.4288

chisq.test(ba1$BA, ba1$Omega3)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$Omega3
## X-squared = 0, df = 1, p-value = 1

chisq.test(ba1$BA, ba1$BMICd)

##
## Pearson's Chi-squared test
##
## data: ba1$BA and ba1$BMICd
## X-squared = 4.1665, df = 2, p-value = 0.1245

chisq.test(ba1$BA, ba1$SANx_cd)

```

```

## Warning in chisq.test(ba1$BA, ba1$SAnx_cd): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba1$BA and ba1$SAnx_cd
## X-squared = 0.0087451, df = 2, p-value = 0.9956

chisq.test(ba1$BA, ba1$SAnx_cd, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$SAnx_cd
## X-squared = 0.0087451, df = NA, p-value = 1

chisq.test(ba1$BA, ba1$LE_cd)

## Warning in chisq.test(ba1$BA, ba1$LE_cd): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$LE_cd
## X-squared = 1.5904, df = 1, p-value = 0.2073

chisq.test(ba1$BA, ba1$LE_cd, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$LE_cd
## X-squared = 3.2101, df = NA, p-value = 0.0714

chisq.test(ba1$BA, ba1$FamilyPast)

## Warning in chisq.test(ba1$BA, ba1$FamilyPast): Chi-squared approximation may
## be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$FamilyPast
## X-squared = 0.97414, df = 1, p-value = 0.3236

chisq.test(ba1$BA, ba1$FamilyPast, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)

```

```
##
## data:  ba1$BA and ba1$FamilyPast
## X-squared = 1.8258, df = NA, p-value = 0.2909

#chisq.test(ba1$BA, ba1$Childhod)
#chisq.test(ba1$BA, ba1$Childhod, simulate.p.value = TRUE)

#j45.1
#сравнение BA0 и BA1
#критерий хи-квадрат
#Heredity, Educ, Chem
print(table(ba1$BA,ba1$Heredity))

##
##      0  1
##  0 45  3
##  1 27 15

print(table(ba1$BA,ba1$Sex))

##
##      0  1
##  0 28 20
##  1 33  9

print(table(ba1$BA,ba1$Family))

##
##      1  2  3  4  5  6
##  0 20  2  1  0  5 20
##  1  2  2  1  2  1  9

print(table(ba1$BA,ba1$Child))

##
##      0  1  2  3  4  5
##  0 25  5 16  1  1  0
##  1  4  4  6  2  0  1

print(table(ba1$BA,ba1$Educ))

##
##      2  3  4  5  6
##  0  6  3  7 29  3
##  1  0  5  3  8  1

print(table(ba1$BA,ba1$ScaleS))

##
##      1  3  4  5  6  7  8  9 10
##  0  0  0  5 10 10 12 10  1  0
##  1  0  3  2  3  3  4  1  1  0

print(table(ba1$BA,ba1$ProfCateg))
```



```

##
##      1  2  3  4  5
##    0  0 12 19 13  3
##    1  0 10  5  9  8

print(table(ba1$BA,ba1$DustCurrent))

##
##      0  1
##    0 36 12
##    1 18  1

print(table(ba1$BA,ba1$BioCurrent))

##
##      0  1
##    0 48  0
##    1 16  3

chisq.test(ba2$BA, ba2$Atopia)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba2$BA and ba2$Atopia
## X-squared = 82.027, df = 1, p-value < 2.2e-16

chisq.test(ba2$BA, ba2$Heredity)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba2$BA and ba2$Heredity
## X-squared = 27.165, df = 1, p-value = 1.869e-07

chisq.test(ba2$BA, ba2$Sex)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba2$BA and ba2$Sex
## X-squared = 0, df = 1, p-value = 1

chisq.test(ba2$BA, ba2$Family)

## Warning in chisq.test(ba2$BA, ba2$Family): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data:  ba2$BA and ba2$Family
## X-squared = 1.5424, df = 4, p-value = 0.8191

chisq.test(ba2$BA, ba2$Family, simulate.p.value = TRUE)

```

```

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$Family
## X-squared = 1.5424, df = NA, p-value = 0.8626

chisq.test(ba2$BA, ba2$FamType)

## Warning in chisq.test(ba2$BA, ba2$FamType): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba2$BA and ba2$FamType
## X-squared = 2.6954, df = 4, p-value = 0.61

chisq.test(ba2$BA, ba2$FamType, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$FamType
## X-squared = 2.6954, df = NA, p-value = 0.6357

chisq.test(ba2$BA, ba2$Child)

## Warning in chisq.test(ba2$BA, ba2$Child): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba2$BA and ba2$Child
## X-squared = 2.3895, df = 4, p-value = 0.6645

chisq.test(ba2$BA, ba2$Child, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$Child
## X-squared = 2.3895, df = NA, p-value = 0.7006

chisq.test(ba2$BA, ba2$Child_8)

## Warning in chisq.test(ba2$BA, ba2$Child_8): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##

```

```

## data: ba2$BA and ba2$Child_8
## X-squared = 6.8967e-31, df = 1, p-value = 1

chisq.test(ba2$BA, ba2$Child_8, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$Child_8
## X-squared = 0.02374, df = NA, p-value = 1

chisq.test(ba1$BA, ba1$Hobby)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba1$BA and ba1$Hobby
## X-squared = 1.8274, df = 1, p-value = 0.1764

chisq.test(ba1$BA, ba1$Educ)

## Warning in chisq.test(ba1$BA, ba1$Educ): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba1$BA and ba1$Educ
## X-squared = 8.0698, df = 4, p-value = 0.08905

chisq.test(ba1$BA, ba1$Educ, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba1$BA and ba1$Educ
## X-squared = 8.0698, df = NA, p-value = 0.08146

chisq.test(ba2$BA, ba2$ScaleS)

## Warning in chisq.test(ba2$BA, ba2$ScaleS): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba2$BA and ba2$ScaleS
## X-squared = 6.2999, df = 6, p-value = 0.3904

chisq.test(ba2$BA, ba2$ScaleS, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000

```

```

## replicates)
##
## data: ba2$BA and ba2$ScaleS
## X-squared = 6.2999, df = NA, p-value = 0.4113

chisq.test(ba2$BA, ba2$ScaleR)

## Warning in chisq.test(ba2$BA, ba2$ScaleR): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba2$BA and ba2$ScaleR
## X-squared = 8.3042, df = 9, p-value = 0.5038

chisq.test(ba2$BA, ba2$ScaleR, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$ScaleR
## X-squared = 8.3042, df = NA, p-value = 0.5287

chisq.test(ba2$BA, ba2$ISL_cd)

## Warning in chisq.test(ba2$BA, ba2$ISL_cd): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba2$BA and ba2$ISL_cd
## X-squared = 1.4287, df = 3, p-value = 0.6988

chisq.test(ba2$BA, ba2$ISL_cd, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$ISL_cd
## X-squared = 1.4287, df = NA, p-value = 0.7661

chisq.test(ba2$BA, ba2$ProfCateg)

## Warning in chisq.test(ba2$BA, ba2$ProfCateg): Chi-squared approximation may
be
## incorrect

##
## Pearson's Chi-squared test
##

```

```

## data: ba2$BA and ba2$ProfCateg
## X-squared = 4.3668, df = 4, p-value = 0.3586

chisq.test(ba2$BA, ba2$ProfCateg, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$ProfCateg
## X-squared = 4.3668, df = NA, p-value = 0.3598

chisq.test(ba2$BA, ba2$DustCurrent)

## Warning in chisq.test(ba2$BA, ba2$DustCurrent): Chi-squared approximation may
be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba2$BA and ba2$DustCurrent
## X-squared = 1.1337, df = 1, p-value = 0.287

chisq.test(ba2$BA, ba2$DustCurrent, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$DustCurrent
## X-squared = 1.9429, df = NA, p-value = 0.2069

chisq.test(ba2$BA, ba2$Dust)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba2$BA and ba2$Dust
## X-squared = 0.0022135, df = 1, p-value = 0.9625

chisq.test(ba2$BA, ba2$ChemCurrent)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba2$BA and ba2$ChemCurrent
## X-squared = 0.00042427, df = 1, p-value = 0.9836

chisq.test(ba2$BA, ba2$Chem)

##
## Pearson's Chi-squared test with Yates' continuity correction
##

```

```

## data:  ba2$BA and ba2$Chem
## X-squared = 2.6614, df = 1, p-value = 0.1028

chisq.test(ba2$BA, ba2$Bio)

## Warning in chisq.test(ba2$BA, ba2$Bio): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba2$BA and ba2$Bio
## X-squared = 0.23954, df = 1, p-value = 0.6245

chisq.test(ba2$BA, ba2$Bio, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data:  ba2$BA and ba2$Bio
## X-squared = 0.85995, df = NA, p-value = 0.4503

chisq.test(ba2$BA, ba2$Cold)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba2$BA and ba2$Cold
## X-squared = 0.8075, df = 1, p-value = 0.3689

chisq.test(ba2$BA, ba2$Heat)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba2$BA and ba2$Heat
## X-squared = 1.2568, df = 1, p-value = 0.2623

chisq.test(ba2$BA, ba2$PhysLoad)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data:  ba2$BA and ba2$PhysLoad
## X-squared = 1.5766e-30, df = 1, p-value = 1

chisq.test(ba2$BA, ba2$SMK)

## Warning in chisq.test(ba2$BA, ba2$SMK): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##

```

```

## data: ba2$BA and ba2$SMK
## X-squared = 0.96875, df = 2, p-value = 0.6161

chisq.test(ba2$BA, ba2$SMK, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$SMK
## X-squared = 0.96875, df = NA, p-value = 1

chisq.test(ba2$BA, ba2$FruVegs)

## Warning in chisq.test(ba2$BA, ba2$FruVegs): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba2$BA and ba2$FruVegs
## X-squared = 1.2635e-31, df = 1, p-value = 1

chisq.test(ba2$BA, ba2$FruVegs, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$FruVegs
## X-squared = 0.00775, df = NA, p-value = 1

chisq.test(ba2$BA, ba2$Omega3)

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba2$BA and ba2$Omega3
## X-squared = 0.14222, df = 1, p-value = 0.7061

chisq.test(ba2$BA, ba2$SAnx_cd)

## Warning in chisq.test(ba2$BA, ba2$SAnx_cd): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba2$BA and ba2$SAnx_cd
## X-squared = 0.30987, df = 2, p-value = 0.8565

chisq.test(ba2$BA, ba2$SAnx_cd, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000

```

```

## replicates)
##
## data: ba2$BA and ba2$SAnx_cd
## X-squared = 0.30987, df = NA, p-value = 0.9035

chisq.test(ba2$BA, ba2$LE_cd)

## Warning in chisq.test(ba2$BA, ba2$LE_cd): Chi-squared approximation may be
## incorrect

##
## Pearson's Chi-squared test
##
## data: ba2$BA and ba2$LE_cd
## X-squared = 12.961, df = 2, p-value = 0.001533

chisq.test(ba2$BA, ba2$LE_cd, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$LE_cd
## X-squared = 12.961, df = NA, p-value = 0.002499

chisq.test(ba2$BA, ba2$FamilyPast)

## Warning in chisq.test(ba2$BA, ba2$FamilyPast): Chi-squared approximation may
be
## incorrect

##
## Pearson's Chi-squared test with Yates' continuity correction
##
## data: ba2$BA and ba2$FamilyPast
## X-squared = 0.015976, df = 1, p-value = 0.8994

chisq.test(ba2$BA, ba2$FamilyPast, simulate.p.value = TRUE)

##
## Pearson's Chi-squared test with simulated p-value (based on 2000
## replicates)
##
## data: ba2$BA and ba2$FamilyPast
## X-squared = 0.23388, df = NA, p-value = 0.7271

#chisq.test(ba2$BA, ba2$Chldhod)
#tchisq.test(ba2$BA, ba2$Chldhod, simulate.p.value = TRUE)

#j45.0
#сравнение BA0 и BA1
#критерий хи-квадрат

```



```
#Atopia, Heredity, Educ, Chem
print(table(ba2$BA,ba2$Atopia))
```

```
##
##      0  1
##    0 46  0
##    1  0 40
```

```
print(table(ba2$BA,ba2$Heredity))
```

```
##
##      0  1
##    0 45  3
##    1 16 24
```

```
print(table(ba2$BA,ba2$Educ))
```

```
##
##      2  3  4  5  6
##    0  6  3  7 29  3
##    1  4  2  1 10  0
```

```
print(table(ba2$BA,ba2$Chem))
```

```
##
##      0  1
##    0 21 26
##    1 14  6
```

```
"Моделирование зависимости ВА от РМ"
```

```
## [1] "Моделирование зависимости ВА от РМ"
```

```
#Связь РМ и БА неаллергический фенотип
#PM2.5Avr и TBLogAvr - ОШ примерно 3!!!
```

```
m1 <- glm(BA ~ PM2.5MaxAvr + BMI + Age + DL + ERI_Ef_Ph + ERI_Ef_M + Overcom,
          family = 'binomial', data = ba1)
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(m1)
```

```
##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + BMI + Age + DL + ERI_Ef_Ph +
##      ERI_Ef_M + Overcom, family = "binomial", data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.881e-05 -2.110e-08 -2.110e-08 -2.110e-08  2.001e-05
##
## Coefficients:
```

```

##           Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.953e+02  6.271e+05  -0.001    0.999
## PM2.5MaxAvr  1.152e+03  1.281e+06   0.001    0.999
## BMI          5.664e+00  1.302e+04   0.000    1.000
## Age         -2.358e-01  4.279e+03   0.000    1.000
## DL          -7.436e-01  6.754e+03   0.000    1.000
## ERI_Ef_Ph    2.568e-01  3.695e+04   0.000    1.000
## ERI_Ef_M     1.679e+00  4.810e+04   0.000    1.000
## Overcom     1.140e+01  2.353e+04   0.000    1.000
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1.8597e+01  on 25  degrees of freedom
## Residual deviance: 1.4853e-09  on 18  degrees of freedom
## (64 observations deleted due to missingness)
## AIC: 16
##
## Number of Fisher Scoring iterations: 25

m1.1 <- glm(BA ~ PM2.5MaxAvr + BMI + Age + DL + ERI_Ef_Ph + ERI_Ef_M,
            family = 'binomial', data = ba1)

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(m1.1)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + BMI + Age + DL + ERI_Ef_Ph +
##      ERI_Ef_M, family = "binomial", data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.996e-05 -2.110e-08 -2.110e-08 -2.110e-08  2.069e-05
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.694e+02  4.233e+05  -0.001    0.999
## PM2.5MaxAvr  1.221e+03  1.238e+06   0.001    0.999
## BMI          4.292e+00  9.524e+03   0.000    1.000
## Age          1.830e+00  3.194e+03   0.001    1.000
## DL          -2.779e+00  5.785e+03   0.000    1.000
## ERI_Ef_Ph    1.153e+00  3.256e+04   0.000    1.000
## ERI_Ef_M     1.139e+01  3.963e+04   0.000    1.000
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1.8597e+01  on 25  degrees of freedom
## Residual deviance: 1.7518e-09  on 19  degrees of freedom
## (64 observations deleted due to missingness)
## AIC: 14

```

```
##
## Number of Fisher Scoring iterations: 25

m1.2 <- glm(BA ~ PM2.5MaxAvr + BMI + Age + DL,
            family = 'binomial', data = ba1)

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(m1.2)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + BMI + Age + DL, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.96725  -0.00333   0.00000   0.00000   2.11555
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -159.9660    148.7195  -1.076   0.282
## PM2.5MaxAvr  288.6127    268.8857   1.073   0.283
## BMI           1.2781     1.1141   1.147   0.251
## Age           0.3335     0.3501   0.953   0.341
## DL            0.8118     0.7790   1.042   0.297
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 23.5605  on 29  degrees of freedom
## Residual deviance:  9.1802  on 25  degrees of freedom
## (60 observations deleted due to missingness)
## AIC: 19.18
##
## Number of Fisher Scoring iterations: 11

m1.3 <- glm(BA ~ PM2.5MaxAvr + BMI + Age,
            family = 'binomial', data = ba1)

summary(m1.3)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + BMI + Age, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.6920  -0.7195  -0.5153   0.9677   2.0202
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -6.05884    2.03830  -2.972  0.00295 **
## PM2.5MaxAvr  15.98647    8.00355   1.997  0.04578 *
## BMI           0.03553    0.05658   0.628  0.52997
```

```
## Age          0.04458    0.02258    1.974    0.04838 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 60.751  on 57  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 68.751
##
## Number of Fisher Scoring iterations: 4

m1.4 <- glm(BA ~ PM2.5MaxAvr + Age,
            family = 'binomial', data = ba1)
summary(m1.4)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Age, family = "binomial", data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.5638  -0.7087  -0.5100   0.9104   2.0144
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.36026    1.68022  -3.190  0.00142 **
## PM2.5MaxAvr 16.03541    7.98776   2.007  0.04470 *
## Age          0.04932    0.02126   2.320  0.02033 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 61.148  on 58  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 67.148
##
## Number of Fisher Scoring iterations: 4

m2 <- glm(BA ~ PM2.5MaxAvr + Atopia, family = 'binomial', data = ba1)
summary(m2)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Atopia, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.9922  -0.8155  -0.6468   0.6876   2.1352
```

```
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.902      1.321  -2.197   0.028 *
## PM2.5MaxAvr   12.339      8.275   1.491   0.136
## Atopia1       16.720    1455.398   0.011   0.991
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 68.963  on 58  degrees of freedom
## Residual deviance: 63.728  on 56  degrees of freedom
## (31 observations deleted due to missingness)
## AIC: 69.728
##
## Number of Fisher Scoring iterations: 14

m2.1 <- glm(BA ~ PM2.5MaxAvr + Heredity, family = 'binomial', data = ba1)
summary(m2.1)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.8521  -0.7180  -0.5161   0.7361   2.3558
##
## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.4906      1.3932  -2.505   0.01223 *
## PM2.5MaxAvr  13.1781      8.6178   1.529   0.12622
## Heredity1     2.3915      0.8092   2.955   0.00312 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 57.059  on 58  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 63.059
##
## Number of Fisher Scoring iterations: 4

m2.2 <- glm(BA ~ PM2.5MaxAvr + Sex, family = 'binomial', data = ba1)
summary(m2.2)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Sex, family = "binomial", data = ba1)
```

```
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -1.107   -0.835   -0.631    1.250    2.019
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.6351     1.2950  -2.035  0.0419 *
## PM2.5MaxAvr  12.4264     8.1002   1.534  0.1250
## Sex1         -0.8637     0.7251  -1.191  0.2336
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 65.423  on 58  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 71.423
##
## Number of Fisher Scoring iterations: 4

m2.3 <- glm(BA ~ PM2.5MaxAvr + Family, family = 'binomial', data = ba1)
summary(m2.3)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Family, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -0.9456   -0.4893   -0.4397   -0.3818    2.1477
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.1033     1.8371  -1.689  0.0912 .
## PM2.5MaxAvr   5.2441    10.8232   0.485  0.6280
## Family2       1.4893     1.4506   1.027  0.3046
## Family3     -13.7685    2399.5451  -0.006  0.9954
## Family5       1.0989     1.3948   0.788  0.4308
## Family6       0.2153     1.0724   0.201  0.8409
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 36.945  on 50  degrees of freedom
## Residual deviance: 34.988  on 45  degrees of freedom
## (39 observations deleted due to missingness)
## AIC: 46.988
##
## Number of Fisher Scoring iterations: 15
```

```
m2.4 <- glm(BA ~ PM2.5MaxAvr + Child, family = 'binomial', data = ba1)
summary(m2.4)
```

```
##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Child, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.6621  -0.5486  -0.4684  -0.4047   2.3722
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.1804     1.7064  -1.864   0.0623 .
## PM2.5MaxAvr   7.2405     10.2114   0.709   0.4783
## Child1        0.6171     1.2782   0.483   0.6292
## Child2        0.2969     0.9960   0.298   0.7656
## Child3       -15.6310    3956.1804  -0.004   0.9968
## Child4       -15.2328    3956.1804  -0.004   0.9969
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 36.945  on 50  degrees of freedom
## Residual deviance: 35.767  on 45  degrees of freedom
## (39 observations deleted due to missingness)
## AIC: 47.767
##
## Number of Fisher Scoring iterations: 16
```

```
m2.5 <- glm(BA ~ PM2.5MaxAvr + Educ, family = 'binomial', data = ba1)
summary(m2.5)
```

```
##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Educ, family = "binomial", data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.83012  -0.56966  -0.48976  -0.00008   2.18538
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -20.1431    4364.2945  -0.005   0.996
## PM2.5MaxAvr   4.9225     11.0914   0.444   0.657
## Educ3        18.3442    4364.2944   0.004   0.997
## Educ4       -0.1126    5949.0626   0.000   1.000
## Educ5        17.5601    4364.2943   0.004   0.997
## Educ6        18.3388    4364.2944   0.004   0.997
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
## Null deviance: 36.945 on 50 degrees of freedom
## Residual deviance: 32.352 on 45 degrees of freedom
## (39 observations deleted due to missingness)
## AIC: 44.352
##
## Number of Fisher Scoring iterations: 18

m2.6 <- glm(BA ~ PM2.5MaxAvr + ScaleS, family = 'binomial', data = ba1)
summary(m2.6)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + ScaleS, family = "binomial",
## data = ba1)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -0.89599 -0.60631 -0.41687 -0.00007 2.17110
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.0210 1.9156 -1.055 0.291
## PM2.5MaxAvr 7.6487 11.6952 0.654 0.513
## ScaleS5 -18.6206 3530.7516 -0.005 0.996
## ScaleS6 -0.6891 1.1478 -0.600 0.548
## ScaleS7 -0.8243 1.1421 -0.722 0.470
## ScaleS8 -18.6743 3562.3132 -0.005 0.996
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 36.945 on 50 degrees of freedom
## Residual deviance: 30.210 on 45 degrees of freedom
## (39 observations deleted due to missingness)
## AIC: 42.21
##
## Number of Fisher Scoring iterations: 18

m2.7 <- glm(BA ~ PM2.5MaxAvr + DustCurrent, family = 'binomial', data = ba1)
summary(m2.7)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + DustCurrent, family = "binomial",
## data = ba1)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -0.92607 -0.64680 -0.40489 -0.00013 1.95643
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.572 2.153 -2.123 0.0337 *
```



```
## PM2.5MaxAvr    19.886    13.320    1.493    0.1355
## DustCurrent1  -17.144   1953.631   -0.009    0.9930
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 41.087  on 51  degrees of freedom
## Residual deviance: 34.894  on 49  degrees of freedom
## (38 observations deleted due to missingness)
## AIC: 40.894
##
## Number of Fisher Scoring iterations: 17

m2.8 <- glm(BA ~ PM2.5MaxAvr + BioCurrent, family = 'binomial', data = ba1)
summary(m2.8)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + BioCurrent, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.6975  -0.5858  -0.4768  -0.2649   2.1123
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -4.182      1.980  -2.112  0.0347 *
## PM2.5MaxAvr   14.571      12.233   1.191  0.2336
## BioCurrent1   18.242     2399.545   0.008  0.9939
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 41.087  on 51  degrees of freedom
## Residual deviance: 35.216  on 49  degrees of freedom
## (38 observations deleted due to missingness)
## AIC: 41.216
##
## Number of Fisher Scoring iterations: 15

m2.9 <- glm(BA ~ PM2.5MaxAvr + ProfCateg, family = 'binomial', data = ba1)
summary(m2.9)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + ProfCateg, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -1.4863 -0.6553 -0.4783 -0.2946 2.1094
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.2161      1.3934 -2.308  0.0210 *
## PM2.5MaxAvr 15.2737      8.4185  1.814  0.0696 .
## ProfCateg3  -1.5214      0.9705 -1.568  0.1170
## ProfCateg4   0.2515      0.8702  0.289  0.7726
## ProfCateg5   0.8863      1.0559  0.839  0.4013
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 61.210  on 56  degrees of freedom
## Residual deviance: 52.481  on 52  degrees of freedom
## (33 observations deleted due to missingness)
## AIC: 62.481
##
## Number of Fisher Scoring iterations: 5

m3 <- glm(BA ~ PM2.5MaxAvr + Age + Heredity,
          family = 'binomial', data = ba1)
summary(m3)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Age + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1370  -0.6232  -0.4298   0.5349   2.0665
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.49308    1.84411 -2.979  0.00289 **
## PM2.5MaxAvr 15.75423    8.72360  1.806  0.07093 .
## Age          0.04188    0.02323  1.802  0.07148 .
## Heredity1    2.14925    0.82811  2.595  0.00945 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 53.714  on 57  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 61.714
##
## Number of Fisher Scoring iterations: 5
```

```

m3.1 <- glm(BA ~ PM2.5MaxAvr + Heredity,
             family = 'binomial', data = ba1)
summary(m3.1)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.8521  -0.7180  -0.5161   0.7361   2.3558
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.4906     1.3932  -2.505  0.01223 *
## PM2.5MaxAvr  13.1781     8.6178   1.529  0.12622
## Heredity1     2.3915     0.8092   2.955  0.00312 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 57.059  on 58  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 63.059
##
## Number of Fisher Scoring iterations: 4

anova(m3,m3.1,test = "Chisq")

## Analysis of Deviance Table
##
## Model 1: BA ~ PM2.5MaxAvr + Age + Heredity
## Model 2: BA ~ PM2.5MaxAvr + Heredity
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1         57      53.714
## 2         58      57.059 -1   -3.3453  0.0674 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

m4 <- glm(BA ~ PM2.5MaxAvr + Age + Heredity,
           family = 'binomial', data = ba1)
summary(m4)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Age + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max

```

```
## -2.1370 -0.6232 -0.4298 0.5349 2.0665
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.49308    1.84411  -2.979  0.00289 **
## PM2.5MaxAvr 15.75423    8.72360   1.806  0.07093 .
## Age          0.04188    0.02323   1.802  0.07148 .
## Heredity1    2.14925    0.82811   2.595  0.00945 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 53.714  on 57  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 61.714
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m4, incr = list(PM2.5MaxAvr = 0.01, Age = 1,
                                           Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1 PM2.5MaxAvr    1.171      1.000      1.419      0.01
## 2 Age            1.043      0.997      1.094      1
## 3 Heredity1      8.578      1.814     50.775 Indicator variable

m5 <- glm(BA ~ PM10MaxAvr + Age + Heredity,
          family = 'binomial', data = ba1)
summary(m5)

##
## Call:
## glm(formula = BA ~ PM10MaxAvr + Age + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1698  -0.6135  -0.4024   0.4993   2.1786
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.36746    2.01791  -3.155  0.0016 **
## PM10MaxAvr  11.00857    5.09908   2.159  0.0309 *
## Age          0.04001    0.02342   1.708  0.0875 .
## Heredity1    2.15576    0.84855   2.541  0.0111 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
## Null deviance: 70.204 on 60 degrees of freedom
## Residual deviance: 52.062 on 57 degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 60.062
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m5, incr = list(PM10MaxAvr = 0.01, Age = 1,
                                           Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 PM10MaxAvr 1.116 1.017 1.248 0.01
## 2 Age 1.041 0.995 1.093 1
## 3 Heredity1 8.634 1.765 53.710 Indicator variable

m6 <- glm(BA ~ TSPMaxAvr + Age + Heredity,
          family = 'binomial', data = ba1)
summary(m6)

##
## Call:
## glm(formula = BA ~ TSPMaxAvr + Age + Heredity, family = "binomial",
## data = ba1)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -2.0803 -0.5712 -0.4495 0.5358 2.4426
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -7.97574 2.85608 -2.793 0.00523 **
## TSPMaxAvr 8.73882 4.46281 1.958 0.05021 .
## Age 0.04336 0.02344 1.849 0.06440 .
## Heredity1 2.04813 0.82696 2.477 0.01326 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 70.204 on 60 degrees of freedom
## Residual deviance: 52.778 on 57 degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 60.778
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m6, incr = list(TSPMaxAvr = 0.01, Age = 1,
                                           Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 TSPMaxAvr 1.091 1.008 1.207 0.01
```

```
## 2      Age      1.044      0.998      1.096      1
## 3 Heredity1    7.753      1.635     45.575 Indicator variable
```

```
m7 <- glm(BA ~ PM2.5Avr + Age + Heredity,
          family = 'binomial', data = ba1)
summary(m7)
```

```
##
## Call:
## glm(formula = BA ~ PM2.5Avr + Age + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0605  -0.6221  -0.4146   0.4213   2.2925
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -6.53084    1.98150  -3.296 0.000981 ***
## PM2.5Avr     128.02235    54.32984   2.356 0.018453 *
## Age           0.04639     0.02412   1.923 0.054490 .
## Heredity1     2.16356     0.88540   2.444 0.014542 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 50.003  on 57  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 58.003
##
## Number of Fisher Scoring iterations: 5
```

```
# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m7, incr = list(PM2.5Avr = 0.01, Age = 1,
                                           Heredity = 1 ))
```

```
## predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1 PM2.5Avr      3.597      1.397      12.685      0.01
## 2      Age      1.047      1.000      1.102      1
## 3 Heredity1     8.702      1.687     60.597 Indicator variable
```

```
m8 <- glm(BA ~ PM10Avr + Age + Heredity,
          family = 'binomial', data = ba1)
summary(m8)
```

```
##
## Call:
## glm(formula = BA ~ PM10Avr + Age + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -2.0593 -0.6357 -0.3894 0.4183 2.4515
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -7.63426    2.33705  -3.267  0.00109 **
## PM10Avr      56.33754   23.25631   2.422  0.01542 *
## Age          0.04353    0.02393   1.819  0.06890 .
## Heredity1    2.12567    0.89272   2.381  0.01726 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 70.204 on 60 degrees of freedom
## Residual deviance: 49.430 on 57 degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 57.43
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m8, incr = list(PM10Avr = 0.01, Age = 1,
                                           Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 PM10Avr 1.757 1.174 3.022 0.01
## 2 Age 1.044 0.998 1.098 1
## 3 Heredity1 8.379 1.602 59.425 Indicator variable

m9 <- glm(BA ~ TSPAavr + Age + Heredity,
          family = 'binomial', data = ba1)
summary(m9)

##
## Call:
## glm(formula = BA ~ TSPAavr + Age + Heredity, family = "binomial",
## data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0983 -0.6171 -0.4063  0.4770  2.4211
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.9467    3.0208  -2.962  0.00306 **
## TSPAavr      39.7019   17.8299   2.227  0.02597 *
## Age          0.0410    0.0235   1.745  0.08105 .
## Heredity1    2.0982    0.8590   2.443  0.01458 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##      Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 51.233  on 57  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 59.233
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m9, incr = list(TSPAver = 0.01, Age = 1,
                                           Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1      TSPAver      1.487      1.084      2.229      0.01
## 2      Age      1.042      0.996      1.094      1
## 3 Heredity1      8.152      1.638      52.284 Indicator variable

m10 <- glm(BA ~ TBlogMax + Age + Heredity,
           family = 'binomial', data = ba1)
summary(m10)

##
## Call:
## glm(formula = BA ~ TBlogMax + Age + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.2086  -0.6127  -0.3780   0.4676   2.2060
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -7.13251    2.25912  -3.157  0.00159 **
## TBlogMax      490.70880   218.23323   2.249  0.02454 *
## Age           0.04106    0.02351   1.746  0.08074 .
## Heredity1     2.17986    0.86533   2.519  0.01177 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 51.016  on 57  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 59.016
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m10, incr = list(TBlogMax = 0.001, Age = 1,
                                           Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1      TBlogMax      1.633      1.111      2.676      0.001
```



```
## 2      Age      1.042      0.996      1.094      1
## 3 Heredity1    8.845      1.768      58.039 Indicator variable

m11 <- glm(BA ~ PLogMax + Age + Heredity,
           family = 'binomial', data = ba1)
summary(m11)

##
## Call:
## glm(formula = BA ~ PLogMax + Age + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1536  -0.6183  -0.3753   0.4523   2.3124
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -6.88256    2.15188  -3.198  0.00138 **
## PLogMax       350.72492   155.99013   2.248  0.02455 *
## Age           0.04232    0.02362   1.792  0.07317 .
## Heredity1     2.15556    0.86888   2.481  0.01311 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 70.204  on 60  degrees of freedom
## Residual deviance: 50.836  on 57  degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 58.836
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m11, incr = list(PLogMax = 0.001, Age = 1,
                                             Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1      PLogMax      1.420      1.081      2.030      0.001
## 2         Age      1.043      0.997      1.096      1
## 3 Heredity1      8.633      1.713      57.200 Indicator variable

m12 <- glm(BA ~ TBPLogMax + Age + Heredity,
           family = 'binomial', data = ba1)
summary(m12)

##
## Call:
## glm(formula = BA ~ TBPLogMax + Age + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -2.1778 -0.6191 -0.3769 0.4581 2.2706
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.99894    2.20000  -3.181  0.00147 **
## TBPLogMax    205.15241   91.17992   2.250  0.02445 *
## Age           0.04181    0.02358   1.773  0.07618 .
## Heredity1     2.16575    0.86772   2.496  0.01256 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 70.204 on 60 degrees of freedom
## Residual deviance: 50.894 on 57 degrees of freedom
## (29 observations deleted due to missingness)
## AIC: 58.894
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m12, incr = list(TBPLogMax = 0.001, Age = 1,
                                             Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 TBPLogMax 1.228 1.046 1.511 0.001
## 2 Age 1.043 0.996 1.095 1
## 3 Heredity1 8.721 1.735 57.611 Indicator variable

m13 <- glm(BA ~ TBLogAvr + Age + Heredity,
            family = 'binomial', data = ba1)
summary(m13)

##
## Call:
## glm(formula = BA ~ TBLogAvr + Age + Heredity, family = "binomial",
## data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0848  -0.5937  -0.4178   0.5218   2.3407
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.47857    2.02196  -3.204  0.00135 **
## TBLogAvr    1124.06212  509.38129   2.207  0.02733 *
## Age           0.04380    0.02346   1.867  0.06185 .
## Heredity1     2.14391    0.87277   2.456  0.01403 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
## Null deviance: 69.590 on 59 degrees of freedom
## Residual deviance: 50.486 on 56 degrees of freedom
## (30 observations deleted due to missingness)
## AIC: 58.486
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m13, incr = list(TBLogAvr = 0.001, Age = 1,
                                             Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 TBLogAvr 3.077 1.273 10.185 0.001
## 2 Age 1.045 0.999 1.097 1
## 3 Heredity1 8.533 1.690 57.657 Indicator variable

m14 <- glm(BA ~ PLogAvr + Age + Heredity,
            family = 'binomial', data = ba1)
summary(m14)

##
## Call:
## glm(formula = BA ~ PLogAvr + Age + Heredity, family = "binomial",
## data = ba1)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -2.1103 -0.6013 -0.3489 0.4930 2.3111
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.58101 2.02149 -3.256 0.00113 **
## PLogAvr 899.83200 385.77565 2.333 0.01967 *
## Age 0.04307 0.02359 1.826 0.06790 .
## Heredity1 2.20886 0.89581 2.466 0.01367 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 69.590 on 59 degrees of freedom
## Residual deviance: 49.542 on 56 degrees of freedom
## (30 observations deleted due to missingness)
## AIC: 57.542
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m14, incr = list(PLogAvr = 0.001, Age = 1,
                                             Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 PLogAvr 2.459 1.263 6.069 0.001
```

```
## 2      Age      1.044      0.998      1.096      1
## 3 Heredity1    9.105      1.751      66.261 Indicator variable

m15 <- glm(BA ~ TBPLogAvr + Age + Heredity,
           family = 'binomial', data = ba1)
summary(m15)

##
## Call:
## glm(formula = BA ~ TBPLogAvr + Age + Heredity, family = "binomial",
##      data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1001  -0.6057  -0.3827   0.5044   2.3469
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -6.35752    1.94134  -3.275  0.00106 **
## TBPLogAvr     474.33953   207.84423   2.282  0.02248 *
## Age           0.04388    0.02353   1.865  0.06222 .
## Heredity1     2.17474    0.88505   2.457  0.01400 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 69.590  on 59  degrees of freedom
## Residual deviance: 49.901  on 56  degrees of freedom
## (30 observations deleted due to missingness)
## AIC: 57.901
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba1, model = m15, incr = list(TBPLogAvr = 0.001, Age = 1,
                                             Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1 TBPLogAvr      1.607      1.122      2.621      0.001
## 2      Age      1.045      0.999      1.097      1
## 3 Heredity1      8.800      1.715      61.861 Indicator variable

"Моделирование зависимости БА от РМ"

## [1] "Моделирование зависимости БА от РМ"

#Связь РМ и БА аллергический фенотип
#Статистически достоверная роль пассивного курения!!!, но не РМ

m1 <- glm(BA ~ PM2.5MaxAvr + PasSMK,
          family = 'binomial', data = ba2)
summary(m1)
```

```
##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + PasSMK, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -2.1845   -0.7796   -0.3693    0.5780    1.4491
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.6933     2.1060  -1.754   0.0795 .
## PM2.5MaxAvr  17.8722    13.2128   1.353   0.1762
## PasSMK        1.2176     0.6131   1.986   0.0470 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 21.334  on 20  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 27.334
##
## Number of Fisher Scoring iterations: 5

m2 <- glm(BA ~ PM2.5MaxAvr + Atopia, family = 'binomial', data = ba2)
summary(m2)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Atopia, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -2.409e-06  -2.409e-06  -2.409e-06   2.409e-06   2.409e-06
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.657e+01  1.700e+05   0.000     1
## PM2.5MaxAvr -9.946e-13  1.159e+06   0.000     1
## Atopia1      5.313e+01  9.882e+04   0.001     1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 8.1004e+01  on 63  degrees of freedom
## Residual deviance: 3.7130e-10  on 61  degrees of freedom
## (24 observations deleted due to missingness)
## AIC: 6
##
## Number of Fisher Scoring iterations: 25
```

```
m2.1 <- glm(BA ~ PM2.5MaxAvr + Heredity, family = 'binomial', data = ba2)
summary(m2.1)
```

```
##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1386  -0.6734  -0.4265   0.5810   2.2103
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -4.2999     1.5996  -2.688 0.007187 **
## PM2.5MaxAvr  18.3443     9.7441   1.883 0.059755 .
## Heredity1     2.8383     0.7739   3.668 0.000245 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 82.565  on 65  degrees of freedom
## Residual deviance: 58.215  on 63  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 64.215
##
## Number of Fisher Scoring iterations: 5
```

```
m2.2 <- glm(BA ~ PM2.5MaxAvr + Educ, family = 'binomial', data = ba2)
summary(m2.5)
```

```
##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Educ, family = "binomial", data = ba1)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.83012  -0.56966  -0.48976  -0.00008   2.18538
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -20.1431  4364.2945  -0.005   0.996
## PM2.5MaxAvr    4.9225   11.0914   0.444   0.657
## Educ3         18.3442  4364.2944   0.004   0.997
## Educ4         -0.1126  5949.0626   0.000   1.000
## Educ5         17.5601  4364.2943   0.004   0.997
## Educ6         18.3388  4364.2944   0.004   0.997
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 36.945  on 50  degrees of freedom
## Residual deviance: 32.352  on 45  degrees of freedom
```

```
## (39 observations deleted due to missingness)
## AIC: 44.352
##
## Number of Fisher Scoring iterations: 18

m2.3 <- glm(BA ~ PM2.5MaxAvr + Chem, family = 'binomial', data = ba2)
summary(m2.3)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Chem, family = "binomial", data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.1879  -0.6262  -0.4098  -0.1500   2.3618
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -4.1544     2.0205  -2.056  0.0398 *
## PM2.5MaxAvr  21.0537    12.4505   1.691  0.0908 .
## Chem1        -1.5545     0.8816  -1.763  0.0778 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 51.750  on 53  degrees of freedom
## Residual deviance: 44.107  on 51  degrees of freedom
## (34 observations deleted due to missingness)
## AIC: 50.107
##
## Number of Fisher Scoring iterations: 5

m2.4 <- glm(BA ~ PM2.5MaxAvr + Heredity + Chem, family = 'binomial', data = ba2)
summary(m2.4)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + Heredity + Chem, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.36090  -0.52795  -0.25051  -0.09352   2.88485
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -4.527     2.402  -1.885  0.05948 .
## PM2.5MaxAvr  18.535    14.570   1.272  0.20332
## Heredity1     3.514     1.187   2.960  0.00308 **
## Chem1        -2.245     1.227  -1.829  0.06733 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 51.750 on 53 degrees of freedom
## Residual deviance: 31.112 on 50 degrees of freedom
## (34 observations deleted due to missingness)
## AIC: 39.112
##
## Number of Fisher Scoring iterations: 6

m3 <- glm(BA ~ PM2.5MaxAvr + PasSMK + Heredity + Chem,
          family = 'binomial', data = ba2)

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(m3)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + PasSMK + Heredity + Chem, family =
"binomial",
## data = ba2)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -1.195e-05 -2.110e-08 -2.110e-08 -2.110e-08 1.039e-05
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -17.26 144125.36 0.000 1
## PM2.5MaxAvr -121.30 911509.63 0.000 1
## PasSMK 40.12 75227.44 0.001 1
## Heredity1 62.53 146661.51 0.000 1
## Chem1 -62.25 142277.83 0.000 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 1.5442e+01 on 15 degrees of freedom
## Residual deviance: 4.2639e-10 on 11 degrees of freedom
## (72 observations deleted due to missingness)
## AIC: 10
##
## Number of Fisher Scoring iterations: 25

m3.1 <- glm(BA ~ PM2.5MaxAvr + PasSMK + Heredity,
            family = 'binomial', data = ba2)
summary(m3.1)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + PasSMK + Heredity, family = "binomial",
## data = ba2)
##
## Deviance Residuals:
```



```
##      Min      1Q   Median      3Q      Max
## -1.9634 -0.5029 -0.2841  0.6501  1.9774
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.9006      2.2012  -1.772  0.0764 .
## PM2.5MaxAvr  12.1980     13.8109   0.883  0.3771
## PasSMK        1.6247      0.7399   2.196  0.0281 *
## Heredity1     2.0197      1.3678   1.477  0.1398
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 18.818  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 26.818
##
## Number of Fisher Scoring iterations: 5

m4 <- glm(BA ~ PM2.5MaxAvr + PasSMK + Heredity,
          family = 'binomial', data = ba2)
summary(m4)

##
## Call:
## glm(formula = BA ~ PM2.5MaxAvr + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min      1Q   Median      3Q      Max
## -1.9634 -0.5029 -0.2841  0.6501  1.9774
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.9006      2.2012  -1.772  0.0764 .
## PM2.5MaxAvr  12.1980     13.8109   0.883  0.3771
## PasSMK        1.6247      0.7399   2.196  0.0281 *
## Heredity1     2.0197      1.3678   1.477  0.1398
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 18.818  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 26.818
##
## Number of Fisher Scoring iterations: 5
```

```

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m4, incr = list(PM2.5MaxAvr = 0.01, PasSMK = 1,
                                           Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1 PM2.5MaxAvr      1.130      0.876      1.581      0.01
## 2      PasSMK      5.077      1.500      31.919      1
## 3      Heredity1      7.536      0.635      200.780 Indicator variable

m5 <- glm(BA ~ PM10MaxAvr + PasSMK + Heredity,
          family = 'binomial', data = ba2)
summary(m5)

##
## Call:
## glm(formula = BA ~ PM10MaxAvr + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0827  -0.4441  -0.2701   0.5735   1.8200
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.0845     2.6312  -1.932   0.0533 .
## PM10MaxAvr    10.4453     8.4254   1.240   0.2151
## PasSMK         1.5830     0.7525   2.104   0.0354 *
## Heredity1      1.8345     1.4053   1.305   0.1917
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 17.998  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 25.998
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m5, incr = list(PM10MaxAvr = 0.01, PasSMK = 1,
                                           Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1 PM10MaxAvr      1.110      0.949      1.346      0.01
## 2      PasSMK      4.870      1.395      31.334      1
## 3      Heredity1      6.262      0.462      173.599 Indicator variable

m6 <- glm(BA ~ TSPMaxAvr + PasSMK + Heredity,
          family = 'binomial', data = ba2)
summary(m6)

```

```
##
## Call:
## glm(formula = BA ~ TSPMaxAvr + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0253  -0.4297  -0.1536   0.6435   1.7690
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -9.2627     5.6957  -1.626   0.1039
## TSPMaxAvr     13.3278    10.0208   1.330   0.1835
## PasSMK         1.3997     0.7468   1.874   0.0609 .
## Heredity1      1.8711     1.4366   1.302   0.1928
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 17.009  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 25.009
##
## Number of Fisher Scoring iterations: 6

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m6, incr = list(TSPMaxAvr = 0.01, PasSMK = 1,
                                           Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1 TSPMaxAvr      1.143      0.978      1.496      0.01
## 2 PasSMK         4.054      1.158      25.663      1
## 3 Heredity1      6.495      0.460     192.824 Indicator variable

m7 <- glm(BA ~ PM2.5Avr + PasSMK + Heredity,
          family = 'binomial', data = ba2)
summary(m7)

##
## Call:
## glm(formula = BA ~ PM2.5Avr + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0856  -0.4961  -0.1773   0.4256   1.6102
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.6534     2.6819  -2.108   0.035 *
## PM2.5Avr     150.8600    97.3175   1.550   0.121
```

```

## PasSMK      1.6664      0.8275      2.014      0.044 *
## Heredity1    1.6211      1.4391      1.126      0.260
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 16.330  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 24.33
##
## Number of Fisher Scoring iterations: 6

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m7, incr = list(PM2.5Avr = 0.01, PasSMK = 1,
                                           Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1  PM2.5Avr      4.520      0.909      55.678      0.01
## 2   PasSMK      5.293      1.396      45.884      1
## 3 Heredity1      5.059      0.350      146.358 Indicator variable

m8 <- glm(BA ~ PM10Avr + PasSMK + Heredity,
          family = 'binomial', data = ba2)
summary(m8)

##
## Call:
## glm(formula = BA ~ PM10Avr + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0711  -0.5139  -0.1772   0.4321   1.6414
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -6.7597     3.4149  -1.979   0.0478 *
## PM10Avr       60.7844    40.2702   1.509   0.1312
## PasSMK        1.6850     0.8238   2.045   0.0408 *
## Heredity1     1.6637     1.4253   1.167   0.2431
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 16.483  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 24.483
##
## Number of Fisher Scoring iterations: 6

```

```

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m8, incr = list(PM10Avr = 0.01, PasSMK = 1,
                                           Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 PM10Avr 1.836 0.949 5.265 0.01
## 2 PasSMK 5.393 1.435 46.752 1
## 3 Heredity1 5.279 0.382 150.425 Indicator variable

m9 <- glm(BA ~ TSPAavr + PasSMK + Heredity,
          family = 'binomial', data = ba2)
summary(m9)

##
## Call:
## glm(formula = BA ~ TSPAavr + PasSMK + Heredity, family = "binomial",
## data = ba2)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -2.0671 -0.5060 -0.2273 0.5208 1.7648
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.0028 4.5701 -1.751 0.0799 .
## TSPAavr 40.7453 30.2658 1.346 0.1782
## PasSMK 1.6250 0.7769 2.092 0.0365 *
## Heredity1 1.7817 1.4048 1.268 0.2047
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 30.789 on 22 degrees of freedom
## Residual deviance: 17.488 on 19 degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 25.488
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m9, incr = list(TSPAavr = 0.01, PasSMK = 1,
                                           Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 TSPAavr 1.503 0.883 3.161 0.01
## 2 PasSMK 5.078 1.414 35.269 1
## 3 Heredity1 5.940 0.449 164.600 Indicator variable

m10 <- glm(BA ~ TBLogMax + PasSMK + Heredity,
           family = 'binomial', data = ba2)
summary(m10)

```

```
##
## Call:
## glm(formula = BA ~ TBlogMax + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0129  -0.5116  -0.2399   0.5904   1.8849
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.087      2.886  -1.762   0.078 .
## TBlogMax     368.593    330.026   1.117   0.264
## PasSMK        1.625      0.744   2.184   0.029 *
## Heredity1     1.929      1.372   1.406   0.160
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 18.189  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 26.189
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m10, incr = list(TBlogMax = 0.001, PasSMK = 1,
                                             Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1  TBlogMax      1.446      0.811      3.322      0.001
## 2   PasSMK      5.077      1.493     32.306      1
## 3 Heredity1      6.880      0.579    184.360 Indicator variable

m11 <- glm(BA ~ PLogMax + PasSMK + Heredity,
           family = 'binomial', data = ba2)
summary(m11)

##
## Call:
## glm(formula = BA ~ PLogMax + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0263  -0.5127  -0.1971   0.5473   1.8113
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.4508      3.0032  -1.815   0.0695 .
## PLogMax      319.2318    257.9393   1.238   0.2159
```

```

## PasSMK      1.6239      0.7515      2.161      0.0307 *
## Heredity1    1.8445      1.3894      1.328      0.1843
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 17.615  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 25.615
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m11, incr = list(PLogMax = 0.001, PasSMK = 1,
                                             Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1      PLogMax      1.376      0.901      2.746      0.001
## 2      PasSMK      5.073      1.482      33.275      1
## 3 Heredity1      6.325      0.512      173.559 Indicator variable

m12 <- glm(BA ~ TBPLogMax + PasSMK + Heredity,
            family = 'binomial', data = ba2)
summary(m12)

##
## Call:
## glm(formula = BA ~ TBPLogMax + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0220  -0.5288  -0.2150   0.5661   1.8423
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.3006     2.9443  -1.800   0.0718 .
## TBPLogMax    172.2510    144.3646   1.193   0.2328
## PasSMK        1.6224     0.7476   2.170   0.0300 *
## Heredity1     1.8799     1.3816   1.361   0.1736
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 17.863  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 25.863
##
## Number of Fisher Scoring iterations: 5

```

```

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m12, incr = list(TBLogMax = 0.001, PasSMK = 1,
                                             Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 TBLogMax      1.188      0.931      1.732      0.001
## 2 PasSMK        5.065      1.485      32.686      1
## 3 Heredity1     6.553      0.540     177.872 Indicator variable

m13 <- glm(BA ~ TBLogAvr + PasSMK + Heredity,
           family = 'binomial', data = ba2)
summary(m13)

##
## Call:
## glm(formula = BA ~ TBLogAvr + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0389  -0.4819  -0.2025   0.5153   1.7589
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.3160     2.6477  -2.008  0.0447 *
## TBLogAvr     1157.8213    831.7571   1.392  0.1639
## PasSMK        1.6298     0.7932   2.055  0.0399 *
## Heredity1     1.8155     1.3979   1.299  0.1940
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 17.203  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 25.203
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m13, incr = list(TBLogAvr = 0.001, PasSMK = 1,
                                             Heredity = 1 ))

## predictor oddsratio ci_low (2.5) ci_high (97.5) increment
## 1 TBLogAvr      3.183      0.769      26.265      0.001
## 2 PasSMK        5.103      1.382      37.179      1
## 3 Heredity1     6.144      0.480     169.279 Indicator variable

m14 <- glm(BA ~ PLogAvr + PasSMK + Heredity,
           family = 'binomial', data = ba2)
summary(m14)

```



```
##
## Call:
## glm(formula = BA ~ PLogAvr + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0790  -0.4884  -0.2414   0.4762   1.7177
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.0362     2.2933  -2.196   0.0281 *
## PLogAvr       836.2378    555.1412   1.506   0.1320
## PasSMK         1.6676     0.8166   2.042   0.0411 *
## Heredity1      1.7876     1.4017   1.275   0.2022
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 17.026  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 25.026
##
## Number of Fisher Scoring iterations: 5

# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m14, incr = list(PLogAvr = 0.001, PasSMK = 1,
                                             Heredity = 1 ))

##      predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1   PLogAvr      2.308      0.849      8.781      0.001
## 2   PasSMK      5.300      1.386     41.512      1
## 3 Heredity1      5.975      0.454    164.685 Indicator variable

m15 <- glm(BA ~ TBPLogAvr + PasSMK + Heredity,
            family = 'binomial', data = ba2)
summary(m15)

##
## Call:
## glm(formula = BA ~ TBPLogAvr + PasSMK + Heredity, family = "binomial",
##      data = ba2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0573  -0.4657  -0.2048   0.4953   1.7224
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.0727     2.3973  -2.116   0.0343 *
## TBPLogAvr    478.5812    325.6433   1.470   0.1417
```

```
## PasSMK      1.6312      0.8046      2.027      0.0426 *
## Heredity1    1.7875      1.4050      1.272      0.2033
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 30.789  on 22  degrees of freedom
## Residual deviance: 16.961  on 19  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 24.961
##
## Number of Fisher Scoring iterations: 5
```

```
# Calculate OR for specific increment step of continuous variable
or_glm(data = ba2, model = m15, incr = list(TBPLogAvr = 0.001, PasSMK = 1,
                                             Heredity = 1 ))
```

```
## predictor oddsratio ci_low (2.5) ci_high (97.5)      increment
## 1 TBPLogAvr      1.614      0.919      3.641      0.001
## 2 PasSMK         5.110      1.360      38.546      1
## 3 Heredity1      5.974      0.456      165.972 Indicator variable
```

```
"Цитокины и БА"
```

```
## [1] "Цитокины и БА"
```

```
#Еос выше при увеличении степени тяжести, ниже у j45.1
```

```
#ИЛ6 и ИЛ10 выше у j45.1 (тенденция, p<0.1)
```

```
bacit <- ba[ba$BAcd %in% c("j45.0", "j45.1"), ]
str(bacit)
```

```
## 'data.frame':      82 obs. of  106 variables:
## $ ID              : chr  "3" "60" "74" "52" ...
## $ Kazan           : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 2 2 1 1 ...
## $ Point           : Factor w/ 10 levels "1","4","5","6",...: NA NA 1 NA NA NA
## 10 1 NA NA ...
## $ TSPAвр          : num  NA NA 0.164 NA NA NA 0.15 0.164 NA NA ...
## $ TSPMaxAvр        : num  NA NA 0.595 NA NA ...
## $ TSPMaxMax        : num  NA NA 0.734 NA NA NA 0.72 0.734 NA NA ...
## $ PM10Avр          : num  NA NA 0.094 NA NA NA 0.081 0.094 NA NA ...
## $ PM10MaxAvр       : num  NA NA 0.348 NA NA ...
## $ PM10MaxMax       : num  NA NA 0.44 NA NA NA 0.42 0.44 NA NA ...
## $ PM2.5Avр         : num  NA NA 0.031 NA NA NA 0.025 0.031 NA NA ...
## $ PM2.5MaxAvр      : num  NA NA 0.172 NA NA ...
## $ PM2.5MaxMax      : num  NA NA 0.253 NA NA NA 0.379 0.253 NA NA ...
## $ TSPDT            : num  NA NA 0.038 NA NA NA 0.058 0.038 NA NA ...
## $ PM10DT           : num  NA NA 0.03 NA NA NA 0.046 0.03 NA NA ...
## $ PM25DT           : num  NA NA 0.028 NA NA NA 0.026 0.028 NA NA ...
## $ Cu               : num  NA NA 0.56 NA NA NA NA 0.56 NA NA ...
## $ C                : num  NA NA 93.4 NA NA ...
## $ LAL1             : num  NA NA 0.0277 NA NA NA 0.0064 0.0277 NA NA ...
## $ LAL2             : num  NA NA 0.0694 NA NA NA 0.0382 0.0694 NA NA ...
## $ TBPLogMax        : num  NA NA 0.0221 NA NA ...
```

```

## $ PLogMax      : num  NA NA 0.0126 NA NA ...
## $ TBlogMax     : num  NA NA 0.00948 NA NA ...
## $ TBPLogAvr    : num  NA NA 0.00804 NA NA ...
## $ PLogAvr      : num  NA NA 0.00457 NA NA ...
## $ TBlogAvr     : num  NA NA 0.00346 NA NA ...
## $ BA           : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
## $ BAcid        : Factor w/ 4 levels "j45.0","j45.1",...: 1 2 2 1 2 1 1 2 2
1 ...
## $ BMI          : num  27.5 39 39.1 25.9 22.7 ...
## $ BMICd        : Factor w/ 3 levels "1","2","3": 2 3 3 2 1 2 1 2 2 NA ...
## $ BAsevere     : Factor w/ 3 levels "1","2","3": 2 3 2 2 3 3 1 2 2 3 ...
## $ BAobstr      : Factor w/ 2 levels "0","1": 2 1 2 1 2 2 1 1 1 2 ...
## $ BAcontrol    : Factor w/ 4 levels "1","2","3","4": 2 2 2 NA NA 3 1 3 2 3
...
## $ BAdebut      : Factor w/ 2 levels "0","1": 2 2 1 2 1 1 1 1 1 1 ...
## $ IgE          : num  155 72.9 11.3 NA NA ...
## $ Eos          : num  276 291.2 286 510.3 32.4 ...
## $ Atopia       : Factor w/ 2 levels "0","1": 2 1 1 2 1 2 2 1 1 2 ...
## $ Heredity     : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 1 2 1 ...
## $ Age          : num  39.7 56.7 37.7 38.8 58.7 ...
## $ Sex          : Factor w/ 2 levels "0","1": 2 1 1 1 2 2 1 1 1 1 ...
## $ Family       : Factor w/ 6 levels "1","2","3","4",...: 6 6 6 6 6 6 1 2 6
5 ...
## $ FamType      : Factor w/ 5 levels "0","1","2","3",...: 4 4 4 4 4 2 4 3 4
2 ...
## $ Child        : Factor w/ 6 levels "0","1","2","3",...: 2 3 2 2 2 3 1 3 4
3 ...
## $ Child_8      : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 1 1 2 1 ...
## $ Hobby        : Factor w/ 2 levels "0","1": 1 1 2 1 2 2 2 1 2 1 ...
## $ Educ         : Factor w/ 5 levels "2","3","4","5",...: 4 4 4 4 3 1 4 2 2
2 ...
## $ EducYrs      : num  19 15 16 17 13 11 17 13 13 ...
## $ ScaleS       : Factor w/ 9 levels "1","3","4","5",...: 7 7 6 6 6 6 7 3 4
4 ...
## $ ScaleR       : Factor w/ 10 levels "1","2","3","4",...: 6 5 3 5 7 7 4 4 5
5 ...
## $ ISL_cd       : Factor w/ 4 levels "1","2","3","4": 4 3 2 3 2 2 3 3 3 2
...
## $ ProfCateg    : Factor w/ 5 levels "1","2","3","4",...: 2 2 NA NA 2 NA 3 4
NA NA ...
## $ DustCurrent  : Factor w/ 2 levels "0","1": 2 NA 1 1 1 2 1 1 2 1 ...
## $ pDustAn      : int  NA 1 NA NA 2 NA NA NA NA ...
## $ Dust         : Factor w/ 2 levels "0","1": 2 1 1 1 2 2 1 2 2 2 ...
## $ ChemCurrent  : Factor w/ 2 levels "0","1": 2 NA 1 2 2 2 2 2 2 2 ...
## $ Chem         : Factor w/ 2 levels "0","1": 2 2 1 2 2 2 2 2 2 2 ...
## $ BioCurrent   : Factor w/ 2 levels "0","1": 1 NA 2 1 1 1 1 1 2 1 ...
## $ Bio          : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 1 1 2 1 ...
## $ ColdCurrent  : Factor w/ 2 levels "0","1": 2 NA 2 2 2 2 2 2 2 2 ...
## $ Cold         : Factor w/ 2 levels "0","1": 2 1 2 2 2 2 2 2 2 2 ...
## $ HeatCurrent  : Factor w/ 2 levels "0","1": 2 NA 1 1 1 2 1 2 2 1 ...
## $ Heat         : Factor w/ 2 levels "0","1": 2 1 1 1 2 2 1 2 2 1 ...
## $ PhysLoadCurrent: Factor w/ 2 levels "0","1": 2 NA 2 2 2 2 2 2 2 2 ...

```

```

## $ PhysLoad      : Factor w/ 2 levels "0","1": 2 1 2 2 2 2 2 2 2 2 ...
## $ DL            : int  74 NA 60 70 66 46 NA 78 60 68 ...
## $ JD            : int  21 NA 23 24 21 27 NA 23 22 26 ...
## $ CS            : int  18 NA 17 20 18 10 NA 18 16 21 ...
## $ SS            : int  13 NA NA 18 17 9 NA 18 15 19 ...
## $ SocSup        : int  31 NA NA 38 35 19 NA 36 31 40 ...
## $ JobDis        : num  0.733 NA 0.1 0 0 ...
## $ JI            : int  14 NA 10 11 12 10 NA 12 11 14 ...
## $ JDJI          : int  35 NA 33 35 33 37 NA 35 33 40 ...
## $ PE            : int  10 NA 11 11 10 16 NA 13 12 10 ...
## $ JCQedcd       : Factor w/ 2 levels "0","1": 1 NA NA 1 1 2 NA 1 1 1 ...
## $ ERI_Ef_Ph     : int  9 NA 10 8 7 20 NA 17 10 9 ...
## $ ERI_Ef_M      : int  8 NA 9 7 6 16 NA 14 9 8 ...
## $ Overcom       : int  11 NA 12 7 13 15 NA 18 12 11 ...
## $ SANx          : int  33 46 42 38 29 53 NA 52 NA 52 ...
## $ SANx_cd       : Factor w/ 3 levels "1","2","3": 2 3 2 2 1 3 NA 3 NA 3 ...
## $ LE            : int  108 53 71 109 133 101 NA 238 178 53 ...
## $ LE_cd         : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 NA 2 1 1 ...
## $ FamilyPast    : Factor w/ 2 levels "2","3": 2 2 2 2 2 2 2 2 2 2 ...
## $ SMK           : Factor w/ 3 levels "0","1","2": 1 1 2 2 1 1 1 1 1 1 ...
## $ PasSMK        : int  0 0 3 4 0 0 3 2 0 0 ...
## $ PARigMin      : int  240 240 360 120 40 180 120 60 NA 0 ...
## $ PAModMin      : int  120 120 420 0 30 180 60 60 NA 120 ...
## $ WalkMin       : int  300 120 1800 240 180 120 120 60 120 180 ...
## $ FruVegs       : int  0 0 0 0 0 0 0 0 0 0 ...
## $ Omega3        : Factor w/ 2 levels "0","1": 1 1 1 1 2 2 2 2 2 2 ...
## $ DPP4          : num  NA NA 926 NA NA ...
## $ TGFb1         : num  NA NA 76425 NA NA ...
## $ TSLP          : num  0.485 NA 4.62 NA NA NA 4.62 4.62 NA NA ...
## $ IL13          : num  415 NA 102 NA NA ...
## $ IL17A         : num  NA NA 5.18 NA NA NA 5.18 5.18 NA NA ...
## $ IL1b          : num  NA NA 2.06 NA NA ...
## $ IL33          : num  14.55 NA 5.65 NA NA ...
## $ IL4           : num  0.005 NA 1.31 NA NA NA 0.005 0.005 NA NA ...
## $ IL5           : num  8.77 NA 22.51 NA NA ...
## $ IL6           : num  NA NA 46 NA NA ...
## $ IL25          : num  0.08 NA 0.025 NA NA NA 0.19 0.025 NA NA ...
## [list output truncated]

m_cit <- glm(IgE ~ BAcD + BMI,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IgE ~ BAcD + BMI, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -381.33  -199.75   -98.02    64.65   2041.95
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)

```

```

## (Intercept) 750.87 222.40 3.376 0.00134 **
## BAcj45.1 17.63 104.67 0.168 0.86686
## BMI -18.73 8.98 -2.086 0.04157 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 134897.2)
##
## Null deviance: 8214264 on 58 degrees of freedom
## Residual deviance: 7554241 on 56 degrees of freedom
## (23 observations deleted due to missingness)
## AIC: 869.28
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(Eos ~ BAcj + BMI + Sex + BAcvere,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = Eos ~ BAcj + BMI + Sex + BAcvere, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -816.45 -230.70 -99.52 113.91 2660.68
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1272.095 316.628 4.018 0.000146 ***
## BAcj45.1 -231.918 118.789 -1.952 0.054896 .
## BMI -32.479 9.926 -3.272 0.001660 **
## Sex1 -392.275 118.420 -3.313 0.001465 **
## BAcvere2 128.853 219.112 0.588 0.558377
## BAcvere3 504.200 233.401 2.160 0.034181 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 211976)
##
## Null deviance: 21529725 on 75 degrees of freedom
## Residual deviance: 14838318 on 70 degrees of freedom
## (6 observations deleted due to missingness)
## AIC: 1155.5
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(DPP4 ~ BAcj + Sex + BAcvere,
             family = 'gaussian', data = bacit)
summary(m_cit)

```

```
##
## Call:
## glm(formula = DPP4 ~ BAcD + Sex + BAsevere, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -469.92  -132.80    5.62   200.35   498.32
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    877.73     124.44   7.053 1.39e-07 ***
## BAcDj45.1      100.03     107.95   0.927  0.3623
## Sex1           283.94     103.00   2.757  0.0103 *
## BAsevere2       93.93     126.83   0.741  0.4654
## BAsevere3     -416.06     175.01  -2.377  0.0248 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 71677.89)
##
##      Null deviance: 3648464  on 31  degrees of freedom
## Residual deviance: 1935303  on 27  degrees of freedom
## (50 observations deleted due to missingness)
## AIC: 455.13
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TSLP ~ BAcD + Age + BMI + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TSLP ~ BAcD + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -51.38  -20.54  -14.25   11.89  470.26
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -16.7235    47.5795  -0.351   0.727
## BAcDj45.1     33.0864    20.4370   1.619   0.111
## Age           -0.4784     0.7290  -0.656   0.514
## BMI            0.9160     1.8646   0.491   0.625
## Sex1          28.9437    20.2893   1.427   0.160
##
## (Dispersion parameter for gaussian family taken to be 4704.167)
##
##      Null deviance: 271581  on 57  degrees of freedom
## Residual deviance: 249321  on 53  degrees of freedom
```

```
## (24 observations deleted due to missingness)
## AIC: 661.83
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL4 ~ BAcD + BAsevere,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL4 ~ BAcD + BAsevere, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0266  -0.4698  -0.3752   0.0654   4.6431
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.0316     0.5078   2.031  0.0470 *
## BAcDj45.1     0.5352     0.3428   1.561  0.1241
## BAsevere2    -0.6514     0.5503  -1.184  0.2415
## BAsevere3    -1.0921     0.6130  -1.782  0.0802 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.527705)
##
##      Null deviance: 92.176  on 59  degrees of freedom
## Residual deviance: 85.551  on 56  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 201.56
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL5 ~ BAcD + Age + BMI + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL5 ~ BAcD + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -11.462   -7.276   -3.977    0.712   63.505
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.5322    10.2406   0.931  0.356
## BAcDj45.1     5.9298     4.3987   1.348  0.183
## Age          -0.1583     0.1569  -1.009  0.318
```

```
## BMI          0.1651      0.4013    0.411    0.682
## Sex1         -1.0604      4.3669   -0.243    0.809
##
## (Dispersion parameter for gaussian family taken to be 217.9201)
##
##      Null deviance: 12072  on 57  degrees of freedom
## Residual deviance: 11550  on 53  degrees of freedom
## (24 observations deleted due to missingness)
## AIC: 483.65
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL13 ~ BAcD + BAsevere,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL13 ~ BAcD + BAsevere, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -230.36  -133.56   -25.47   106.86   358.06
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    101.72     63.35   1.606   0.1140
## BAcDj45.1      -21.52     42.76  -0.503   0.6168
## BAsevere2       77.00     68.65   1.122   0.2668
## BAsevere3     156.94     76.47   2.052   0.0448 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 23771.77)
##
##      Null deviance: 1451046  on 59  degrees of freedom
## Residual deviance: 1331219  on 56  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 780.71
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL33 ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL33 ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
##  -59.15   -55.11   -20.14    -8.32  1066.07
```



```
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)    21.42     27.10   0.79   0.433
## BAcj45.1       38.50     39.68   0.97   0.336
##
## (Dispersion parameter for gaussian family taken to be 23507.15)
##
## Null deviance: 1385554 on 59 degrees of freedom
## Residual deviance: 1363415 on 58 degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 778.14
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL25 ~ BAcj + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL25 ~ BAcj + Sex, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0192  -0.4393  -0.3103   0.2609  12.1358
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.2359     0.3674  -0.642   0.523
## BAcj45.1      0.5762     0.4461   1.291   0.202
## Sex1          0.7039     0.4619   1.524   0.133
##
## (Dispersion parameter for gaussian family taken to be 2.819918)
##
## Null deviance: 169.95 on 59 degrees of freedom
## Residual deviance: 160.74 on 57 degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 237.4
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TGFb1 ~ BAcj + BAcjsevere,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TGFb1 ~ BAcj + BAcjsevere, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -51202  -6480     566   11090   31410
```

```
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)   76955       7823   9.837 1.38e-10 ***
## BAcadj45.1    11305       7272   1.554  0.1313
## BAsevere2    -17572       8968  -1.959  0.0601 .
## BAsevere3    -25327      12354  -2.050  0.0498 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 358380640)
##
## Null deviance: 1.2092e+10 on 31 degrees of freedom
## Residual deviance: 1.0035e+10 on 28 degrees of freedom
## (50 observations deleted due to missingness)
## AIC: 726.85
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL17A ~ BAc + BMI,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL17A ~ BAc + BMI, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## 8.882e-16  8.882e-16  1.776e-15  2.664e-15  6.217e-15
##
## Coefficients:
##           Estimate Std. Error   t value Pr(>|t|)
## (Intercept)  5.180e+00  1.865e-15  2.778e+15  <2e-16 ***
## BAcadj45.1  -3.440e-16  8.955e-16  -3.840e-01   0.704
## BMI         -1.910e-16  7.686e-17  -2.485e+00   0.019 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 5.630835e-30)
##
## Null deviance: 0.0000e+00 on 31 degrees of freedom
## Residual deviance: 1.6329e-28 on 29 degrees of freedom
## (50 observations deleted due to missingness)
## AIC: -2059.5
##
## Number of Fisher Scoring iterations: 1

m_cit <- glm(IL1b ~ BAc + BAsevere,
             family = 'gaussian', data = bacit)
summary(m_cit)
```

```
##
## Call:
## glm(formula = IL1b ~ BAcD + BAsevere, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.53279 -0.25615 -0.08965  0.19211  2.66395
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.5978     0.2434  10.675 2.23e-11 ***
## BAcDj45.1     0.4483     0.2262   1.981  0.0574 .
## BAsevere2    -0.7249     0.2790  -2.599  0.0148 *
## BAsevere3    -0.8914     0.3843  -2.320  0.0279 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.3467891)
##
##      Null deviance: 12.8708  on 31  degrees of freedom
## Residual deviance:  9.7101  on 28  degrees of freedom
## (50 observations deleted due to missingness)
## AIC: 62.65
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL6 ~ BAcD + Age + BMI,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL6 ~ BAcD + Age + BMI, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -37.425 -18.487 -10.843   8.137 149.977
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -6.7710     30.9667  -0.219  0.8285
## BAcDj45.1    28.1640     16.1045   1.749  0.0913 .
## Age          -0.9791     0.5522  -1.773  0.0871 .
## BMI           2.0133     1.2864   1.565  0.1288
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1460.547)
##
##      Null deviance: 50743  on 31  degrees of freedom
## Residual deviance: 40895  on 28  degrees of freedom
## (50 observations deleted due to missingness)
## AIC: 329.71
```

```
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL4RNA ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL4RNA ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.7410  -0.7410  -0.3622  -0.3622   5.3399
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.3622     0.2973   1.218   0.23
## BAcDj45.1      0.3788     0.4360   0.869   0.39
##
## (Dispersion parameter for gaussian family taken to be 2.033132)
##
##      Null deviance: 84.894  on 42  degrees of freedom
## Residual deviance: 83.358  on 41  degrees of freedom
## (39 observations deleted due to missingness)
## AIC: 156.49
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL17RNA ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL17RNA ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
##  -8.642  -8.642  -0.296  -0.296  164.163
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.2964     5.4871   0.054   0.957
## BAcDj45.1      8.3458     8.0457   1.037   0.306
##
## (Dispersion parameter for gaussian family taken to be 692.5017)
##
##      Null deviance: 29138  on 42  degrees of freedom
## Residual deviance: 28393  on 41  degrees of freedom
## (39 observations deleted due to missingness)
## AIC: 407.21
```

```
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TGFRNA ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TGFRNA ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0958  -2.0958  -0.3950  -0.3283   21.3928
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.0958     0.8756   2.394   0.0213 *
## BAcDj45.1     -1.7009     1.2839  -1.325   0.1926
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 17.6347)
##
##      Null deviance: 753.97  on 42  degrees of freedom
## Residual deviance: 723.02  on 41  degrees of freedom
## (39 observations deleted due to missingness)
## AIC: 249.39
##
## Number of Fisher Scoring iterations: 2

#Аллергический фенотип: выше по сравнению с контролем уровня ИЛ13
bacit <- ba[ba$BAcD %in% c("K", "j45.0"), ]
str(bacit)

## 'data.frame':    88 obs. of  106 variables:
## $ ID              : chr  "3" "52" "81" "86" ...
## $ Kazan           : Factor w/ 2 levels "0","1": 1 1 1 2 1 2 2 2 2 2 ...
## $ Point           : Factor w/ 10 levels "1","4","5","6",...: NA NA NA 10 NA 1
## 1 3 7 3 ...
## $ TSPAвр          : num  NA NA NA 0.15 NA 0.164 0.164 0.107 0.166 0.107 ...
## $ TSPMaxAvр       : num  NA NA NA 0.552 NA ...
## $ TSPMaxMax       : num  NA NA NA 0.72 NA 0.734 0.734 0.495 0.769 0.495 ...
## $ PM10Avр         : num  NA NA NA 0.081 NA 0.094 0.094 0.043 0.089 0.043 ...
## $ PM10MaxAvр      : num  NA NA NA 0.277 NA ...
## $ PM10MaxMax      : num  NA NA NA 0.42 NA 0.44 0.44 0.261 0.45 0.261 ...
## $ PM2.5Avр        : num  NA NA NA 0.025 NA 0.031 0.031 0.01 0.029 0.01 ...
## $ PM2.5MaxAvр     : num  NA NA NA 0.142 NA ...
## $ PM2.5MaxMax     : num  NA NA NA 0.379 NA 0.253 0.253 0.127 0.264 0.127 ...
## $ TSPDT           : num  NA NA NA 0.058 NA 0.038 0.038 0.014 0.061 0.014 ...
## $ PM10DT          : num  NA NA NA 0.046 NA 0.03 0.03 0.011 0.058 0.011 ...
## $ PM25DT          : num  NA NA NA 0.026 NA 0.028 0.028 0.009 0.021 0.009 ...
## $ Cu              : num  NA NA NA NA NA 0.56 0.56 2.29 0.76 2.29 ...
```

```

## $ C : num NA NA NA 77.6 NA ...
## $ LAL1 : num NA NA NA 0.0064 NA 0.0277 0.0277 0.0139 0.0276
0.0139 ...
## $ LAL2 : num NA NA NA 0.0382 NA 0.0694 0.0694 0.0279 0.0553
0.0279 ...
## $ TBPLogMax : num NA NA NA 0.0188 NA ...
## $ PLogMax : num NA NA NA 0.0105 NA ...
## $ TBLogMax : num NA NA NA 0.00829 NA ...
## $ TBPLogAvr : num NA NA NA 0.00759 NA ...
## $ PLogAvr : num NA NA NA 0.00429 NA ...
## $ TBLogAvr : num NA NA NA 0.0033 NA ...
## $ BA : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 1 1 1 ...
## $ BAcid : Factor w/ 4 levels "j45.0","j45.1",...: 1 1 1 1 1 1 1 4 4
4 ...
## $ BMI : num 27.5 25.9 29.4 23.4 NA ...
## $ BMIdc : Factor w/ 3 levels "1","2","3": 2 2 2 1 NA 1 1 1 2 2 ...
## $ BAsevere : Factor w/ 3 levels "1","2","3": 2 2 3 1 3 2 3 NA NA NA
...
## $ BAobstr : Factor w/ 2 levels "0","1": 2 1 2 1 2 2 2 NA NA NA ...
## $ BAcontrol : Factor w/ 4 levels "1","2","3","4": 2 NA 3 1 3 NA NA NA
NA NA ...
## $ BAdebut : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 1 NA NA NA ...
## $ IgE : num 155 NA 51.1 555 NA ...
## $ Eos : num 276 510.3 270 423 97.5 ...
## $ Atopia : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 1 1 1 ...
## $ Heredity : Factor w/ 2 levels "0","1": 2 1 1 1 1 2 1 2 2 1 ...
## $ Age : num 39.7 38.8 46.6 26.4 41.8 ...
## $ Sex : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 2 1 2 ...
## $ Family : Factor w/ 6 levels "1","2","3","4",...: 6 6 6 1 5 1 6 3 6
6 ...
## $ FamType : Factor w/ 5 levels "0","1","2","3",...: 4 4 2 4 2 4 4 3 4
4 ...
## $ Child : Factor w/ 6 levels "0","1","2","3",...: 2 2 3 1 3 1 2 3 3
3 ...
## $ Child_8 : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 1 1 1 1 ...
## $ Hobby : Factor w/ 2 levels "0","1": 1 1 2 2 1 1 1 2 2 1 ...
## $ Educ : Factor w/ 5 levels "2","3","4","5",...: 4 4 1 4 2 1 4 1 3
3 ...
## $ EducYrs : num 19 17 11 17 13 11.5 18 10 13 12 ...
## $ ScaleS : Factor w/ 9 levels "1","3","4","5",...: 7 6 6 7 4 4 6 4 6
4 ...
## $ ScaleR : Factor w/ 10 levels "1","2","3","4",...: 6 5 7 4 5 3 4 5 7
5 ...
## $ ISL_cd : Factor w/ 4 levels "1","2","3","4": 4 3 2 3 2 2 3 3 2 3
...
## $ ProfCateg : Factor w/ 5 levels "1","2","3","4",...: 2 NA NA 3 NA 3 3 4
4 4 ...
## $ DustCurrent : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 2 2 1 ...
## $ pDustAn : int NA NA NA NA NA NA 0 3 3 5 ...
## $ Dust : Factor w/ 2 levels "0","1": 2 1 2 1 2 1 2 2 2 2 ...
## $ ChemCurrent : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 1 2 1 2 ...
## $ Chem : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 1 2 2 2 ...

```

```

## $ BioCurrent      : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ Bio             : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ ColdCurrent     : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 2 2 ...
## $ Cold            : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 2 2 ...
## $ HeatCurrent     : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 2 1 2 ...
## $ Heat            : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 2 1 2 ...
## $ PhysLoadCurrent : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 1 2 ...
## $ PhysLoad        : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 1 2 ...
## $ DL              : int   74 70 46 NA 68 70 52 NA 64 64 ...
## $ JD              : int   21 24 27 NA 26 17 29 NA 21 24 ...
## $ CS              : int   18 20 10 NA 21 17 14 NA 16 16 ...
## $ SS              : int   13 18 9  NA 19 18 14 NA 16 14 ...
## $ SocSup          : int   31 38 19 NA 40 35 28 NA 32 30 ...
## $ JobDis          : num   0.733 0 0.2 NA 0.533 ...
## $ JI              : int   14 11 10 NA 14 17 11 NA NA 12 ...
## $ JDJI           : int   35 35 37 NA 40 34 40 NA NA 36 ...
## $ PE              : int   10 11 16 NA 10 7 11 NA 11 14 ...
## $ JCQedcd         : Factor w/ 2 levels "0","1": 1 1 2 NA 1 1 1 NA 1 1 ...
## $ ERI_Ef_Ph       : int   9 8 20 NA 9 6 6 7 7 7 ...
## $ ERI_Ef_M        : int   8 7 16 NA 8 5 5 5 6 6 ...
## $ Overcom         : int   11 7 15 NA 11 7 13 11 14 9 ...
## $ SANx            : int   33 38 53 NA 52 69 56 44 45 38 ...
## $ SANx_cd         : Factor w/ 3 levels "1","2","3": 2 2 3 NA 3 3 3 2 3 2 ...
## $ LE              : int  108 109 101 NA 53 120 217 100 0 23 ...
## $ LE_cd           : Factor w/ 3 levels "1","2","3": 1 1 1 NA 1 1 2 1 1 1 ...
## $ FamilyPast      : Factor w/ 2 levels "2","3": 2 2 2 2 2 2 2 2 2 2 ...
## $ SMK             : Factor w/ 3 levels "0","1","2": 1 2 1 1 1 1 3 1 1 1 ...
## $ PasSMK          : int    0 4 0 3 0 0 2 0 0 0 ...
## $ PARigMin        : int  240 120 180 120 0 NA NA NA NA 60 ...
## $ PAModMin        : int  120 0 180 60 120 NA NA NA NA 120 ...
## $ WalkMin         : int  300 240 120 120 180 NA NA NA NA 120 ...
## $ FruVegs         : int    0 0 0 0 0 1 1 1 1 0 ...
## $ Omega3          : Factor w/ 2 levels "0","1": 1 1 2 2 2 2 2 2 2 1 ...
## $ DPP4            : num   NA NA NA 1116 NA ...
## $ TGFb1           : num   NA NA NA 57225 NA ...
## $ TSLP            : num   0.485 NA NA 4.62 NA 4.62 4.62 NA 4.62 NA ...
## $ IL13            : num  414.7 NA NA 69.1 NA ...
## $ IL17A           : num   NA NA NA 5.18 NA 5.18 5.18 NA 5.18 NA ...
## $ IL1b            : num   NA NA NA 2.06 NA ...
## $ IL33            : num  14.6 NA NA 200.2 NA ...
## $ IL4             : num   0.005 NA NA 0.005 NA 0.005 0.08 NA 0.03 NA ...
## $ IL5             : num   8.77 NA NA 2.32 NA ...
## $ IL6             : num   NA NA NA 0.505 NA 0.505 4.65 NA 2.92 NA ...
## $ IL25            : num   0.08 NA NA 0.19 NA 0.025 0.025 NA 0.025 NA ...
## [list output truncated]

m_cit <- glm(DPP4 ~ BAcid + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = DPP4 ~ BAcid + Sex, family = "gaussian", data = bacit)

```

```
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -735.35   -224.37    10.09    171.32    596.77
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1010.91      84.83   11.918 3.18e-16 ***
## BAcK         128.78      88.97    1.447   0.154
## Sex1         115.32      86.92    1.327   0.191
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 87542.65)
##
##      Null deviance: 4644595  on 52  degrees of freedom
## Residual deviance: 4377132  on 50  degrees of freedom
## (35 observations deleted due to missingness)
## AIC: 758.45
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TSLP ~ BAcK,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TSLP ~ BAcK, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
##  -34.50   -34.50    -3.05     1.08   1007.88
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.538     22.532   0.157   0.876
## BAcK           35.584     31.175   1.141   0.258
##
## (Dispersion parameter for gaussian family taken to be 16246)
##
##      Null deviance: 1077157  on 66  degrees of freedom
## Residual deviance: 1055990  on 65  degrees of freedom
## (21 observations deleted due to missingness)
## AIC: 843.71
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL4 ~ BAcK + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)
```



```
##
## Call:
## glm(formula = IL4 ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.7830  -0.5040  -0.2442  -0.0145   4.2092
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.985212   0.308912   3.189  0.00221 **
## BAcDK        0.166161   0.226752   0.733  0.46636
## Age         -0.017101   0.007833  -2.183  0.03270 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.8562204)
##
##      Null deviance: 59.186  on 66  degrees of freedom
## Residual deviance: 54.798  on 64  degrees of freedom
## (21 observations deleted due to missingness)
## AIC: 184.67
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL5 ~ BAcD + BMI,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL5 ~ BAcD + BMI, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -8.162  -5.486  -2.804   2.324  43.613
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  20.7424    6.3696   3.256  0.00182 **
## BAcDK        -1.8723    2.3594  -0.794  0.43042
## BMI          -0.5372    0.2541  -2.114  0.03849 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 91.3837)
##
##      Null deviance: 6212.2  on 65  degrees of freedom
## Residual deviance: 5757.2  on 63  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 490.22
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit <- glm(IL13 ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL13 ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -176.00   -65.27   -15.41    44.85   357.60
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   179.18      21.37    8.385 6.05e-12 ***
## BAcDK         -127.08      29.56   -4.298 5.88e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 14611.07)
##
##      Null deviance: 1219687  on 66  degrees of freedom
## Residual deviance: 949719  on 65  degrees of freedom
## (21 observations deleted due to missingness)
## AIC: 836.61
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL33 ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL33 ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -121.62   -68.71   -34.59     3.31  1808.16
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  102.067     79.239    1.288   0.202
## BAcDK         63.214     58.164    1.087   0.281
## Age          -2.411      2.009   -1.200   0.235
##
## (Dispersion parameter for gaussian family taken to be 56337.23)
##
##      Null deviance: 3744674  on 66  degrees of freedom
## Residual deviance: 3605583  on 64  degrees of freedom
## (21 observations deleted due to missingness)
## AIC: 927.99

```

```
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL25 ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL25 ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.1409  -0.1409  -0.0691  -0.0641   3.9941
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.09406    0.09152   1.028   0.308
## BAcDK        0.07179    0.12662   0.567   0.573
##
## (Dispersion parameter for gaussian family taken to be 0.2680207)
##
##      Null deviance: 17.508  on 66  degrees of freedom
## Residual deviance: 17.421  on 65  degrees of freedom
## (21 observations deleted due to missingness)
## AIC: 105.89
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TGFb1 ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TGFb1 ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -60318  -9544    893   14623   34532
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   63834    5256  12.145 <2e-16 ***
## BAcDK         -3368    6468  -0.521   0.605
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 497273777)
##
##      Null deviance: 2.5496e+10  on 52  degrees of freedom
## Residual deviance: 2.5361e+10  on 51  degrees of freedom
## (35 observations deleted due to missingness)
```

```
## AIC: 1215.7
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL17A ~ BAcD + Age + BMI + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL17A ~ BAcD + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.6587  -0.8074  -0.3699   0.1053  15.8612
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.73355     1.88985   3.034  0.00389 **
## BAcDK         0.69586     0.76643   0.908  0.36846
## Age        -0.04399     0.02959  -1.487  0.14363
## BMI         0.05135     0.08873   0.579  0.56548
## Sex1       -0.89286     0.77203  -1.157  0.25320
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6.301532)
##
##      Null deviance: 325.73  on 52  degrees of freedom
## Residual deviance: 302.47  on 48  degrees of freedom
## (35 observations deleted due to missingness)
## AIC: 254.72
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL1b ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL1b ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4718  -0.3808  -0.1131  -0.0070   5.9042
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.396156   0.346458   6.916 8.12e-09 ***
## BAcDK        0.343687   0.280940   1.223   0.227
## Age        -0.011691   0.009386  -1.246   0.219
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.8881286)
##
##      Null deviance: 46.605  on 52  degrees of freedom
## Residual deviance: 44.406  on 50  degrees of freedom
## (35 observations deleted due to missingness)
## AIC: 149.03
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL6 ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL6 ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -33.343  -16.673   -8.363    3.233   136.373
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   34.8425     11.8609   2.938  0.00499 **
## BAcDK         16.0696      9.6179   1.671  0.10101
## Age           -0.8030      0.3213  -2.499  0.01579 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1040.908)
##
##      Null deviance: 59861  on 52  degrees of freedom
## Residual deviance: 52045  on 50  degrees of freedom
## (35 observations deleted due to missingness)
## AIC: 523.56
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL4RNA ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL4RNA ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4468  -0.4468  -0.4468  -0.3622   5.4270
##
```

```
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.36215    0.25103   1.443   0.154
## BAcDK        0.08465    0.31504   0.269   0.789
##
## (Dispersion parameter for gaussian family taken to be 1.449379)
##
##      Null deviance: 88.517  on 62  degrees of freedom
## Residual deviance: 88.412  on 61  degrees of freedom
## (25 observations deleted due to missingness)
## AIC: 206.14
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL17RNA ~ BAcD + Age + BMI + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL17RNA ~ BAcD + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -5.5774  -2.5493  -0.9127   0.5011  29.3237
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.05179    3.68740   1.370   0.1760
## BAcDK        2.07085    1.44087   1.437   0.1560
## Age          0.06919    0.05900   1.173   0.2457
## BMI         -0.35138    0.17146  -2.049   0.0450 *
## Sex1         2.96380    1.43091   2.071   0.0428 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 28.94504)
##
##      Null deviance: 1932.3  on 62  degrees of freedom
## Residual deviance: 1678.8  on 58  degrees of freedom
## (25 observations deleted due to missingness)
## AIC: 397.6
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TGFRNA ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TGFRNA ~ BAcD, family = "gaussian", data = bacit)
```

```
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -2.0958   -1.6452   -1.0829   -0.7281   21.3928
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.0958      0.7997   2.621  0.0111 *
## BAcK          -1.0130      1.0036  -1.009   0.3168
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 14.70947)
##
##      Null deviance: 912.26  on 62  degrees of freedom
## Residual deviance: 897.28  on 61  degrees of freedom
## (25 observations deleted due to missingness)
## AIC: 352.13
##
## Number of Fisher Scoring iterations: 2

#Неаллергический фенотип: ниже DPP4 (тенденция),
#выше по сравнению с контролем уровня ИЛ5, ИЛ13
bacit <- ba[ba$BAcd %in% c("K", "j45.1"), ]
str(bacit)

## 'data.frame':    90 obs. of  106 variables:
## $ ID              : chr  "60" "74" "71" "58" ...
## $ Kazan           : Factor w/ 2 levels "0","1": 1 2 1 2 1 2 2 1 2 2 ...
## $ Point           : Factor w/ 10 levels "1","4","5","6",...: NA 1 NA 1 NA 1 2
## NA 3 7 ...
## $ TSPAvr          : num  NA 0.164 NA 0.164 NA 0.164 0.09 NA 0.107 0.166 ...
## $ TSPMaxAvr       : num  NA 0.595 NA 0.595 NA ...
## $ TSPMaxMax       : num  NA 0.734 NA 0.734 NA 0.734 0.437 NA 0.495 0.769 ...
## $ PM10Avr         : num  NA 0.094 NA 0.094 NA 0.094 0.036 NA 0.043 0.089 ...
## $ PM10MaxAvr      : num  NA 0.348 NA 0.348 NA ...
## $ PM10MaxMax      : num  NA 0.44 NA 0.44 NA 0.44 0.285 NA 0.261 0.45 ...
## $ PM2.5Avr        : num  NA 0.031 NA 0.031 NA 0.031 0.009 NA 0.01 0.029 ...
## $ PM2.5MaxAvr     : num  NA 0.172 NA 0.172 NA ...
## $ PM2.5MaxMax     : num  NA 0.253 NA 0.253 NA 0.253 0.157 NA 0.127 0.264 ...
## $ TSPDT           : num  NA 0.038 NA 0.038 NA 0.038 0.026 NA 0.014 0.061 ...
## $ PM10DT          : num  NA 0.03 NA 0.03 NA 0.03 0.02 NA 0.011 0.058 ...
## $ PM25DT          : num  NA 0.028 NA 0.028 NA 0.028 0.018 NA 0.009 0.021 ...
## $ Cu              : num  NA 0.56 NA 0.56 NA 0.56 NA NA 2.29 0.76 ...
## $ C               : num  NA 93.4 NA 93.4 NA ...
## $ LAL1            : num  NA 0.0277 NA 0.0277 NA 0.0277 NA NA 0.0139 0.0276
## ...
## $ LAL2            : num  NA 0.0694 NA 0.0694 NA 0.0694 NA NA 0.0279 0.0553
## ...
## $ TBPLogMax       : num  NA 0.0221 NA 0.0221 NA ...
## $ PLogMax         : num  NA 0.0126 NA 0.0126 NA ...
## $ TBLogMax        : num  NA 0.00948 NA 0.00948 NA ...
## $ TBPLogAvr       : num  NA 0.00804 NA 0.00804 NA ...
```

```

## $ PLogAvr      : num  NA 0.00457 NA 0.00457 NA ...
## $ TBLogAvr     : num  NA 0.00346 NA 0.00346 NA ...
## $ BA           : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 1 1 ...
## $ BAc4         : Factor w/ 4 levels "j45.0","j45.1",...: 2 2 2 2 2 2 2 2 4
4 ...
## $ BMI          : num  39 39.1 22.7 27.9 26.8 ...
## $ BM1cd        : Factor w/ 3 levels "1","2","3": 3 3 1 2 2 3 2 1 1 2 ...
## $ BAsevere     : Factor w/ 3 levels "1","2","3": 3 2 3 2 2 1 2 2 NA NA ...
## $ BAobstr      : Factor w/ 2 levels "0","1": 1 2 2 1 1 1 1 1 NA NA ...
## $ BAcontrol    : Factor w/ 4 levels "1","2","3","4": 2 2 NA 3 2 NA 1 2 NA
NA ...
## $ BAdebut      : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 2 1 NA NA ...
## $ IgE          : num  72.9 11.3 NA NA NA ...
## $ Eos          : num  291.2 286 32.4 370 54 ...
## $ Atopia       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ Heredity     : Factor w/ 2 levels "0","1": 1 2 1 1 2 1 1 1 2 2 ...
## $ Age          : num  56.7 37.7 58.7 54.7 39.9 ...
## $ Sex          : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 1 2 2 1 ...
## $ Family       : Factor w/ 6 levels "1","2","3","4",...: 6 6 6 2 6 1 5 6 3
6 ...
## $ FamType      : Factor w/ 5 levels "0","1","2","3",...: 4 4 4 3 4 4 4 4 3
4 ...
## $ Child        : Factor w/ 6 levels "0","1","2","3",...: 3 2 2 3 4 1 1 2 3
3 ...
## $ Child_8      : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 1 1 1 ...
## $ Hobby        : Factor w/ 2 levels "0","1": 1 2 2 1 2 1 1 1 2 2 ...
## $ Educ         : Factor w/ 5 levels "2","3","4","5",...: 4 4 3 2 2 4 4 4 1
3 ...
## $ EducYrs      : num  15 16 13 13 13 16 15 17 10 13 ...
## $ ScaleS       : Factor w/ 9 levels "1","3","4","5",...: 7 6 6 3 4 3 5 2 4
6 ...
## $ ScaleR       : Factor w/ 10 levels "1","2","3","4",...: 5 3 7 4 5 5 4 3 5
7 ...
## $ ISL_cd       : Factor w/ 4 levels "1","2","3","4": 3 2 2 3 3 3 2 2 3 2
...
## $ ProfCateg    : Factor w/ 5 levels "1","2","3","4",...: 2 NA 2 4 NA 2 5 NA
4 4 ...
## $ DustCurrent  : Factor w/ 2 levels "0","1": NA 1 1 1 2 1 NA 1 2 2 ...
## $ pDustAn      : int   1 NA 2 NA NA 0 5 5 3 3 ...
## $ Dust         : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 2 2 2 2 ...
## $ ChemCurrent  : Factor w/ 2 levels "0","1": NA 1 2 2 2 1 NA 2 2 1 ...
## $ Chem         : Factor w/ 2 levels "0","1": 2 1 2 2 2 1 2 2 2 2 ...
## $ BioCurrent   : Factor w/ 2 levels "0","1": NA 2 1 1 2 1 NA 1 1 1 ...
## $ Bio          : Factor w/ 2 levels "0","1": 1 2 1 1 2 1 2 1 1 1 ...
## $ ColdCurrent  : Factor w/ 2 levels "0","1": NA 2 2 2 2 1 NA 2 2 2 ...
## $ Cold         : Factor w/ 2 levels "0","1": 1 2 2 2 2 1 2 2 2 2 ...
## $ HeatCurrent  : Factor w/ 2 levels "0","1": NA 1 1 2 2 1 NA 2 2 1 ...
## $ Heat        : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 1 2 2 1 ...
## $ PhysLoadCurrent: Factor w/ 2 levels "0","1": NA 2 2 2 2 1 NA 2 2 1 ...
## $ PhysLoad     : Factor w/ 2 levels "0","1": 1 2 2 2 2 1 2 2 2 1 ...
## $ DL           : int   NA 60 66 78 60 76 NA 58 NA 64 ...
## $ JD           : int   NA 23 21 23 22 23 NA 32 NA 21 ...

```



```
## $ CS : int NA 17 18 18 16 12 NA 17 NA 16 ...
## $ SS : int NA NA 17 18 15 14 NA 7 NA 16 ...
## $ SocSup : int NA NA 35 36 31 26 NA 24 NA 32 ...
## $ JobDis : num NA 0.1 0 0.1 0 ...
## $ JI : int NA 10 12 12 11 16 NA 15 NA NA ...
## $ JDJI : int NA 33 33 35 33 39 NA 47 NA NA ...
## $ PE : int NA 11 10 13 12 8 NA 9 NA 11 ...
## $ JCQedcd : Factor w/ 2 levels "0","1": NA NA 1 1 1 1 NA 1 NA 1 ...
## $ ERI_Ef_Ph : int NA 10 7 17 10 17 NA 20 7 7 ...
## $ ERI_Ef_M : int NA 9 6 14 9 16 NA 19 5 6 ...
## $ Overcom : int NA 12 13 18 12 16 NA 18 11 14 ...
## $ SANx : int 46 42 29 52 NA 42 52 NA 44 45 ...
## $ SANx_cd : Factor w/ 3 levels "1","2","3": 3 2 1 3 NA 2 3 NA 2 3 ...
## $ LE : int 53 71 133 238 178 265 71 237 100 0 ...
## $ LE_cd : Factor w/ 3 levels "1","2","3": 1 1 1 2 1 2 1 2 1 1 ...
## $ FamilyPast : Factor w/ 2 levels "2","3": 2 2 2 2 2 2 2 2 2 2 ...
## $ SMK : Factor w/ 3 levels "0","1","2": 1 2 1 1 1 1 1 2 1 1 ...
## $ PasSMK : int 0 3 0 2 0 0 0 2 0 0 ...
## $ PARigMin : int 240 360 40 60 NA NA 300 0 NA NA ...
## $ PAModMin : int 120 420 30 60 NA NA 120 0 NA NA ...
## $ WalkMin : int 120 1800 180 60 120 NA 240 50 NA NA ...
## $ FruVegs : int 0 0 0 0 0 1 0 0 1 1 ...
## $ Omega3 : Factor w/ 2 levels "0","1": 1 1 2 2 2 2 1 1 2 2 ...
## $ DPP4 : num NA 926 NA 1069 NA ...
## $ TGFb1 : num NA 76425 NA 99988 NA ...
## $ TSLP : num NA 4.62 NA 4.62 NA ...
## $ IL13 : num NA 102.3 NA 40.7 NA ...
## $ IL17A : num NA 5.18 NA 5.18 NA 5.18 5.18 NA NA 5.18 ...
## $ IL1b : num NA 2.06 NA 2.06 NA ...
## $ IL33 : num NA 5.65 NA 1.79 NA ...
## $ IL4 : num NA 1.31 NA 0.005 NA 6.21 0.005 NA NA 0.03 ...
## $ IL5 : num NA 22.51 NA 2.32 NA ...
## $ IL6 : num NA 46 NA 1.4 NA ...
## $ IL25 : num NA 0.025 NA 0.025 NA 0.025 0.025 NA NA 0.025 ...
## [list output truncated]
```

```
m_cit <- glm(DPP4 ~ BAcD + Age + BMI + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)
```

```
##
## Call:
## glm(formula = DPP4 ~ BAcD + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -467.09  -238.75  -16.99   179.61   599.70
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  972.770    229.720   4.235 0.000115 ***
## BAcDK        189.751     98.219   1.932 0.059825 .
```

```
## Age          3.334      3.115    1.070 0.290345
## BMI          -6.339      8.015   -0.791 0.433268
## Sex1         151.980     97.457    1.559 0.126052
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 87818.49)
##
##      Null deviance: 4550705  on 48  degrees of freedom
## Residual deviance: 3864013  on 44  degrees of freedom
## (41 observations deleted due to missingness)
## AIC: 703.55
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TSLP ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TSLP ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -66.51  -47.27  -23.80    3.04   985.21
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  108.005     67.279   1.605   0.114
## BAcDK         -5.781     40.004  -0.145   0.886
## Age          -1.791      1.313  -1.364   0.178
##
## (Dispersion parameter for gaussian family taken to be 21351.61)
##
##      Null deviance: 1324232  on 62  degrees of freedom
## Residual deviance: 1281097  on 60  degrees of freedom
## (27 observations deleted due to missingness)
## AIC: 811.75
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL4 ~ BAcD + Age + BMI,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL4 ~ BAcD + Age + BMI, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2858  -0.6015  -0.2917  -0.0100   4.9910
```

```
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.84617    0.83394   1.015  0.3145
## BAcDK        -0.37305    0.33038  -1.129  0.2635
## Age          -0.02581    0.01129  -2.287  0.0259 *
## BMI           0.04141    0.02964   1.397  0.1676
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.408461)
##
## Null deviance: 90.454  on 61  degrees of freedom
## Residual deviance: 81.691  on 58  degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 203.05
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL5 ~ BAcDK + Age + BMI,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL5 ~ BAcDK + Age + BMI, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -13.034   -6.663   -3.366    1.761   61.797
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11.5478    8.9978   1.283  0.2045
## BAcDK        -7.5509    3.5647  -2.118  0.0384 *
## Age          -0.2410    0.1218  -1.979  0.0526 .
## BMI           0.4450    0.3198   1.391  0.1694
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 163.9631)
##
## Null deviance: 10849.9  on 61  degrees of freedom
## Residual deviance: 9509.9  on 58  degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 497.99
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL13 ~ BAcDK,
             family = 'gaussian', data = bacit)
summary(m_cit)
```

```
##
## Call:
## glm(formula = IL13 ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -178.83   -31.52   -11.81    29.00   268.45
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   191.57      18.75   10.217 7.79e-15 ***
## BAcDK         -139.47      25.15    -5.545 6.71e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 9843.124)
##
##      Null deviance: 903026  on 62  degrees of freedom
## Residual deviance: 600431  on 61  degrees of freedom
## (27 observations deleted due to missingness)
## AIC: 762.01
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL33 ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL33 ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -156.75  -106.23   -50.67     5.91  1783.50
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  263.641    129.239   2.040   0.0458 *
## BAcDK        -29.703     76.845  -0.387   0.7005
## Age          -4.359      2.522  -1.729   0.0890 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 78787.31)
##
##      Null deviance: 4969124  on 62  degrees of freedom
## Residual deviance: 4727239  on 60  degrees of freedom
## (27 observations deleted due to missingness)
## AIC: 894.01
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit <- glm(IL25 ~ BAcD + Age + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL25 ~ BAcD + Age + Sex, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3424  -0.4966  -0.2037  -0.0470  11.9930
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.04890    0.81588   1.286   0.204
## BAcDK         -0.54687    0.46997  -1.164   0.249
## Age          -0.01500    0.01552  -0.967   0.338
## Sex1          0.67353    0.49060   1.373   0.175
##
## (Dispersion parameter for gaussian family taken to be 2.946856)
##
##      Null deviance: 185.05  on 62  degrees of freedom
## Residual deviance: 173.86  on 59  degrees of freedom
## (27 observations deleted due to missingness)
## AIC: 252.74
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TGFb1 ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TGFb1 ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -60318  -6723    3789   14863   34532
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    69727     6258  11.143 8.69e-15 ***
## BAcDK         -9262     7404  -1.251   0.217
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 548204519)
##
##      Null deviance: 2.6623e+10  on 48  degrees of freedom
## Residual deviance: 2.5766e+10  on 47  degrees of freedom
## (41 observations deleted due to missingness)
## AIC: 1129

```

```
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL17A ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL17A ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.1708  -0.9608  -0.4628   0.0115  16.3066
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.41593    1.30985   4.898 1.24e-05 ***
## BAcDK          0.42571    0.85134   0.500   0.619
## Age          -0.02826    0.02537  -1.114   0.271
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6.783528)
##
##      Null deviance: 324.89  on 48  degrees of freedom
## Residual deviance: 312.04  on 46  degrees of freedom
## (41 observations deleted due to missingness)
## AIC: 237.77
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL1b ~ BAcD + Age + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL1b ~ BAcD + Age + Sex, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.7281  -0.4858  -0.1783   0.0090   5.7153
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.14714    0.57083   5.513 1.64e-06 ***
## BAcDK        -0.10705    0.35844  -0.299   0.767
## Age          -0.01662    0.01080  -1.539   0.131
## Sex1         -0.44261    0.35972  -1.230   0.225
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for gaussian family taken to be 1.201163)
##
##      Null deviance: 58.121  on 48  degrees of freedom
## Residual deviance: 54.052  on 45  degrees of freedom
## (41 observations deleted due to missingness)
## AIC: 153.86
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL6 ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL6 ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -51.914  -21.200  -10.940    6.983   156.716
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   73.4954    20.5632   3.574 0.000839 ***
## BAcDK         -18.1689    13.3651  -1.359 0.180641
## Age           -0.9283     0.3983  -2.331 0.024215 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1671.847)
##
##      Null deviance: 87044  on 48  degrees of freedom
## Residual deviance: 76905  on 46  degrees of freedom
## (41 observations deleted due to missingness)
## AIC: 507.62
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL4RNA ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL4RNA ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.7410  -0.7107  -0.4468  -0.4468   5.4270
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.7410     0.3470   2.136  0.0369 *
```

```

## BAcDK          -0.2942      0.4249  -0.692   0.4915
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2.407497)
##
##      Null deviance: 140.79  on 59  degrees of freedom
## Residual deviance: 139.63  on 58  degrees of freedom
## (30 observations deleted due to missingness)
## AIC: 226.95
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL17RNA ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL17RNA ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -8.642  -8.642  -2.122  -2.122  164.163
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    8.642     5.105   1.693  0.0958 .
## BAcDK         -6.520     6.252  -1.043  0.3013
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 521.1474)
##
##      Null deviance: 30793  on 59  degrees of freedom
## Residual deviance: 30227  on 58  degrees of freedom
## (30 observations deleted due to missingness)
## AIC: 549.6
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TGFRNA ~ BAcD + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TGFRNA ~ BAcD + Sex, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3222  -1.0493  -0.5317   0.4093   8.9019
##

```



```
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.5317    0.4201   1.266   0.211
## BAcDK        0.7905    0.5046   1.567   0.123
## Sex1        -0.6836    0.5190  -1.317   0.193
##
## (Dispersion parameter for gaussian family taken to be 3.313515)
##
## Null deviance: 200.93  on 59  degrees of freedom
## Residual deviance: 188.87  on 57  degrees of freedom
## (30 observations deleted due to missingness)
## AIC: 247.08
##
## Number of Fisher Scoring iterations: 2

#TSLP (p<0,1), ИЛ5 (p<0,1), ИЛ6 - выше у неаллергического фенотипа,
#ИЛ13 - у обоих фенотипов, но у j45.1 растет с ростом экспозиции
#TGFRA - выше у j45.1 (тенденция), падает с ростом экспозиции

#Если добавить взаимодействие с экспозицией
#Неаллергическая БА, обусловленная РМ - более низкие уровни ИЛ4
#по сравнению с аллергической БА

bacit <- ba[ba$BAcd %in% c("j45.0", "j45.1", "K"), ]
str(bacit)

## 'data.frame':    130 obs. of  106 variables:
## $ ID             : chr  "3" "60" "74" "52" ...
## $ Kazan          : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 2 2 1 1 ...
## $ Point          : Factor w/ 10 levels "1","4","5","6",...: NA NA 1 NA NA NA
## 10 1 NA NA ...
## $ TSPAvr         : num  NA NA 0.164 NA NA NA 0.15 0.164 NA NA ...
## $ TSPMaxAvr      : num  NA NA 0.595 NA NA ...
## $ TSPMaxMax      : num  NA NA 0.734 NA NA NA 0.72 0.734 NA NA ...
## $ PM10Avr        : num  NA NA 0.094 NA NA NA 0.081 0.094 NA NA ...
## $ PM10MaxAvr     : num  NA NA 0.348 NA NA ...
## $ PM10MaxMax     : num  NA NA 0.44 NA NA NA 0.42 0.44 NA NA ...
## $ PM2.5Avr       : num  NA NA 0.031 NA NA NA 0.025 0.031 NA NA ...
## $ PM2.5MaxAvr    : num  NA NA 0.172 NA NA ...
## $ PM2.5MaxMax    : num  NA NA 0.253 NA NA NA 0.379 0.253 NA NA ...
## $ TSPDT          : num  NA NA 0.038 NA NA NA 0.058 0.038 NA NA ...
## $ PM10DT         : num  NA NA 0.03 NA NA NA 0.046 0.03 NA NA ...
## $ PM25DT         : num  NA NA 0.028 NA NA NA 0.026 0.028 NA NA ...
## $ Cu             : num  NA NA 0.56 NA NA NA NA 0.56 NA NA ...
## $ C              : num  NA NA 93.4 NA NA ...
## $ LAL1           : num  NA NA 0.0277 NA NA NA 0.0064 0.0277 NA NA ...
## $ LAL2           : num  NA NA 0.0694 NA NA NA 0.0382 0.0694 NA NA ...
## $ TBPLogMax      : num  NA NA 0.0221 NA NA ...
## $ PLogMax        : num  NA NA 0.0126 NA NA ...
## $ TBLogMax       : num  NA NA 0.00948 NA NA ...
## $ TBPLogAvr      : num  NA NA 0.00804 NA NA ...
```

```

## $ PLogAvr      : num  NA NA 0.00457 NA NA ...
## $ TBlogAvr    : num  NA NA 0.00346 NA NA ...
## $ BA          : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
## $ BAcid       : Factor w/ 4 levels "j45.0","j45.1",...: 1 2 2 1 2 1 1 2 2
1 ...
## $ BMI         : num  27.5 39 39.1 25.9 22.7 ...
## $ BMICd      : Factor w/ 3 levels "1","2","3": 2 3 3 2 1 2 1 2 2 NA ...
## $ BAsevere    : Factor w/ 3 levels "1","2","3": 2 3 2 2 3 3 1 2 2 3 ...
## $ BAobstr     : Factor w/ 2 levels "0","1": 2 1 2 1 2 2 1 1 1 2 ...
## $ BAcontrol   : Factor w/ 4 levels "1","2","3","4": 2 2 2 NA NA 3 1 3 2 3
...
## $ BAdebut     : Factor w/ 2 levels "0","1": 2 2 1 2 1 1 1 1 1 1 ...
## $ IgE         : num  155 72.9 11.3 NA NA ...
## $ Eos         : num  276 291.2 286 510.3 32.4 ...
## $ Atopia      : Factor w/ 2 levels "0","1": 2 1 1 2 1 2 2 1 1 2 ...
## $ Heredity    : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 1 2 1 ...
## $ Age         : num  39.7 56.7 37.7 38.8 58.7 ...
## $ Sex         : Factor w/ 2 levels "0","1": 2 1 1 1 2 2 1 1 1 1 ...
## $ Family      : Factor w/ 6 levels "1","2","3","4",...: 6 6 6 6 6 6 1 2 6
5 ...
## $ FamType     : Factor w/ 5 levels "0","1","2","3",...: 4 4 4 4 4 2 4 3 4
2 ...
## $ Child       : Factor w/ 6 levels "0","1","2","3",...: 2 3 2 2 2 3 1 3 4
3 ...
## $ Child_8     : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 1 1 2 1 ...
## $ Hobby       : Factor w/ 2 levels "0","1": 1 1 2 1 2 2 2 1 2 1 ...
## $ Educ        : Factor w/ 5 levels "2","3","4","5",...: 4 4 4 4 3 1 4 2 2
2 ...
## $ EducYrs     : num  19 15 16 17 13 11 17 13 13 13 ...
## $ ScaleS      : Factor w/ 9 levels "1","3","4","5",...: 7 7 6 6 6 6 7 3 4
4 ...
## $ ScaleR      : Factor w/ 10 levels "1","2","3","4",...: 6 5 3 5 7 7 4 4 5
5 ...
## $ ISL_cd      : Factor w/ 4 levels "1","2","3","4": 4 3 2 3 2 2 3 3 3 2
...
## $ ProfCateg   : Factor w/ 5 levels "1","2","3","4",...: 2 2 NA NA 2 NA 3 4
NA NA ...
## $ DustCurrent : Factor w/ 2 levels "0","1": 2 NA 1 1 1 2 1 1 2 1 ...
## $ pDustAn     : int  NA 1 NA NA 2 NA NA NA NA NA ...
## $ Dust        : Factor w/ 2 levels "0","1": 2 1 1 1 2 2 1 2 2 2 ...
## $ ChemCurrent : Factor w/ 2 levels "0","1": 2 NA 1 2 2 2 2 2 2 2 ...
## $ Chem        : Factor w/ 2 levels "0","1": 2 2 1 2 2 2 2 2 2 2 ...
## $ BioCurrent  : Factor w/ 2 levels "0","1": 1 NA 2 1 1 1 1 1 2 1 ...
## $ Bio         : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 1 1 2 1 ...
## $ ColdCurrent : Factor w/ 2 levels "0","1": 2 NA 2 2 2 2 2 2 2 2 ...
## $ Cold        : Factor w/ 2 levels "0","1": 2 1 2 2 2 2 2 2 2 2 ...
## $ HeatCurrent : Factor w/ 2 levels "0","1": 2 NA 1 1 1 2 1 2 2 1 ...
## $ Heat        : Factor w/ 2 levels "0","1": 2 1 1 1 2 2 1 2 2 1 ...
## $ PhysLoadCurrent: Factor w/ 2 levels "0","1": 2 NA 2 2 2 2 2 2 2 2 ...
## $ PhysLoad    : Factor w/ 2 levels "0","1": 2 1 2 2 2 2 2 2 2 2 ...
## $ DL          : int  74 NA 60 70 66 46 NA 78 60 68 ...
## $ JD          : int  21 NA 23 24 21 27 NA 23 22 26 ...

```

```
## $ CS : int 18 NA 17 20 18 10 NA 18 16 21 ...
## $ SS : int 13 NA NA 18 17 9 NA 18 15 19 ...
## $ SocSup : int 31 NA NA 38 35 19 NA 36 31 40 ...
## $ JobDis : num 0.733 NA 0.1 0 0 ...
## $ JI : int 14 NA 10 11 12 10 NA 12 11 14 ...
## $ JDJI : int 35 NA 33 35 33 37 NA 35 33 40 ...
## $ PE : int 10 NA 11 11 10 16 NA 13 12 10 ...
## $ JCQedcd : Factor w/ 2 levels "0","1": 1 NA NA 1 1 2 NA 1 1 1 ...
## $ ERI_Ef_Ph : int 9 NA 10 8 7 20 NA 17 10 9 ...
## $ ERI_Ef_M : int 8 NA 9 7 6 16 NA 14 9 8 ...
## $ Overcom : int 11 NA 12 7 13 15 NA 18 12 11 ...
## $ SANx : int 33 46 42 38 29 53 NA 52 NA 52 ...
## $ SANx_cd : Factor w/ 3 levels "1","2","3": 2 3 2 2 1 3 NA 3 NA 3 ...
## $ LE : int 108 53 71 109 133 101 NA 238 178 53 ...
## $ LE_cd : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 NA 2 1 1 ...
## $ FamilyPast : Factor w/ 2 levels "2","3": 2 2 2 2 2 2 2 2 2 ...
## $ SMK : Factor w/ 3 levels "0","1","2": 1 1 2 2 1 1 1 1 1 ...
## $ PasSMK : int 0 0 3 4 0 0 3 2 0 0 ...
## $ PARigMin : int 240 240 360 120 40 180 120 60 NA 0 ...
## $ PAModMin : int 120 120 420 0 30 180 60 60 NA 120 ...
## $ WalkMin : int 300 120 1800 240 180 120 120 60 120 180 ...
## $ FruVegs : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Omega3 : Factor w/ 2 levels "0","1": 1 1 1 1 2 2 2 2 2 ...
## $ DPP4 : num NA NA 926 NA NA ...
## $ TGFb1 : num NA NA 76425 NA NA ...
## $ TSLP : num 0.485 NA 4.62 NA NA NA 4.62 4.62 NA NA ...
## $ IL13 : num 415 NA 102 NA NA ...
## $ IL17A : num NA NA 5.18 NA NA NA 5.18 5.18 NA NA ...
## $ IL1b : num NA NA 2.06 NA NA ...
## $ IL33 : num 14.55 NA 5.65 NA NA ...
## $ IL4 : num 0.005 NA 1.31 NA NA NA 0.005 0.005 NA NA ...
## $ IL5 : num 8.77 NA 22.51 NA NA ...
## $ IL6 : num NA NA 46 NA NA ...
## $ IL25 : num 0.08 NA 0.025 NA NA NA 0.19 0.025 NA NA ...
## [list output truncated]
```

```
m_cit <- glm(DPP4 ~ BAcD + Sex,
              family = 'gaussian', data = bacit)
```

```
summary(m_cit)
```

```
##
```

```
## Call:
```

```
## glm(formula = DPP4 ~ BAcD + Sex, family = "gaussian", data = bacit)
```

```
##
```

```
## Deviance Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -702.83  -224.99  -11.81   186.33   627.15
```

```
##
```

```
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   978.39      84.01  11.646 <2e-16 ***
## BAcDj45.1     -35.54     110.88  -0.320  0.7497
## BAcDK         144.57      90.09   1.605  0.1136
```

```
## Sex1          173.85      80.71   2.154   0.0351 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 90840.55)
##
##      Null deviance: 6539039  on 66  degrees of freedom
## Residual deviance: 5722955  on 63  degrees of freedom
## (63 observations deleted due to missingness)
## AIC: 960.94
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TSLP ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TSLP ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -55.89  -33.79  -14.46   -2.50   992.74
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   43.561     36.048   1.208   0.230
## BAcDj45.1     36.673     32.975   1.112   0.269
## BAcDK          37.723     29.213   1.291   0.200
## Age           -1.196       0.874  -1.369   0.174
##
## (Dispersion parameter for gaussian family taken to be 14224.94)
##
##      Null deviance: 1342385  on 94  degrees of freedom
## Residual deviance: 1294469  on 91  degrees of freedom
## (35 observations deleted due to missingness)
## AIC: 1184
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL4 ~ BAcD * PLogAvr + Age + BMI + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL4 ~ BAcD * PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3918  -0.4656  -0.2366   0.1083   4.7920
```

```
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.298e+00  1.480e+00   1.553  0.12564
## BAcj45.1     -2.326e+00  1.667e+00  -1.395  0.16807
## BAcK         -1.288e+00  1.364e+00  -0.945  0.34859
## PLogAvr      -4.343e+02  3.004e+02  -1.446  0.15341
## Age          -3.024e-02  9.823e-03  -3.079  0.00311 **
## BMI           3.681e-02  2.637e-02   1.396  0.16784
## Sex1          -2.524e-01  2.813e-01  -0.897  0.37317
## BAcj45.1:PLogAvr  7.323e+02  3.968e+02   1.846  0.06979 .
## BAcK:PLogAvr   3.837e+02  3.333e+02   1.151  0.25420
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.03082)
##
## Null deviance: 77.362 on 69 degrees of freedom
## Residual deviance: 62.880 on 61 degrees of freedom
## (60 observations deleted due to missingness)
## AIC: 211.14
##
## Number of Fisher Scoring iterations: 2

coef(m_cit)

##           (Intercept)           BAcj45.1           BAcK           PLogAvr
##           2.29762376          -2.32572737          -1.28819517          -434.30744962
##           Age           BMI           Sex1 BAcj45.1:PLogAvr
##           -0.03024321          0.03681414          -0.25239158          732.30289548
##           BAcK:PLogAvr
##           383.69138692

m_cit <- glm(IL4 ~ BAc * TBLLogAvr + Age + BMI + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL4 ~ BAc * TBLLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3587  -0.4776  -0.2312   0.1012   4.8140
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.653e+00  1.915e+00   1.385  0.17097
## BAcj45.1     -2.679e+00  2.074e+00  -1.292  0.20118
## BAcK         -1.668e+00  1.813e+00  -0.920  0.36111
## TBLLogAvr    -6.779e+02  5.324e+02  -1.273  0.20773
## Age          -2.999e-02  9.928e-03  -3.020  0.00369 **
```

```

## BMI          3.753e-02  2.650e-02   1.416  0.16180
## Sex1         -2.292e-01  2.809e-01  -0.816  0.41765
## BAcj45.1:TBLogAvr 1.056e+03  6.423e+02   1.644  0.10538
## BAcK:TBLogAvr    6.108e+02  5.672e+02   1.077  0.28574
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.041973)
##
##      Null deviance: 77.362  on 69  degrees of freedom
## Residual deviance: 63.560  on 61  degrees of freedom
## (60 observations deleted due to missingness)
## AIC: 211.9
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL5 ~ BAc + Age + BMI + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL5 ~ BAc + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -11.623   -6.248   -3.286    1.260   63.728
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11.0309      6.9298   1.592   0.115
## BAcj45.1      6.0163      3.5755   1.683   0.096 .
## BAcK         -1.5474      3.1539  -0.491   0.625
## Age          -0.1546      0.1027  -1.506   0.136
## BMI           0.1008      0.2798   0.360   0.719
## Sex1         -1.2044      2.9200  -0.412   0.681
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 157.8495)
##
##      Null deviance: 14712  on 92  degrees of freedom
## Residual deviance: 13733  on 87  degrees of freedom
## (37 observations deleted due to missingness)
## AIC: 742.45
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL13 ~ BAc * TBLogAvr + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

```

```
##
## Call:
## glm(formula = IL13 ~ BAcD * TBLogAvr + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -94.92   -39.65   -20.19    16.24   463.44
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      428.20      139.82   3.063  0.00323 **
## BAcDj45.1       -463.35      170.15  -2.723  0.00835 **
## BAcDK           -375.18      147.65  -2.541  0.01353 *
## TBLogAvr       -102498.72  42776.20  -2.396  0.01955 *
## Sex1            -32.39       22.27  -1.454  0.15078
## BAcDj45.1:TBLogAvr 147947.19  52281.72   2.830  0.00624 **
## BAcDK:TBLogAvr    105830.90  46147.52   2.293  0.02518 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7094.775)
##
##      Null deviance: 543649  on 69  degrees of freedom
## Residual deviance: 446971  on 63  degrees of freedom
## (60 observations deleted due to missingness)
## AIC: 827.97
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL33 ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL33 ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -131.69   -74.51   -37.80     0.65   1801.03
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   120.917     69.716   1.734  0.0862 .
## BAcDj45.1     78.008     63.773   1.223  0.2244
## BAcDK         64.221     56.497   1.137  0.2586
## Age           -2.974     1.690  -1.759  0.0819 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 53205.03)
##
```

```
## Null deviance: 5065551 on 94 degrees of freedom
## Residual deviance: 4841657 on 91 degrees of freedom
## (35 observations deleted due to missingness)
## AIC: 1309.3
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL25 ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL25 ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4912  -0.4862  -0.1409  -0.0691  12.6638
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.09406    0.24995   0.376   0.708
## BAcDj45.1     0.42219    0.36589   1.154   0.252
## BAcDK         0.07179    0.34582   0.208   0.836
##
## (Dispersion parameter for gaussian family taken to be 1.999191)
##
## Null deviance: 186.93 on 94 degrees of freedom
## Residual deviance: 183.93 on 92 degrees of freedom
## (35 observations deleted due to missingness)
## AIC: 340.36
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TGFb1 ~ BAcD,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = TGFb1 ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -60318  -7315      913   14555   34532
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   63834      5227  12.213 <2e-16 ***
## BAcDj45.1     5894      7902   0.746   0.459
## BAcDK        -3368      6432  -0.524   0.602
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



```
##
## (Dispersion parameter for gaussian family taken to be 491759279)
##
## Null deviance: 3.2337e+10 on 66 degrees of freedom
## Residual deviance: 3.1473e+10 on 64 degrees of freedom
## (63 observations deleted due to missingness)
## AIC: 1536
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL17A ~ BAcD + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL17A ~ BAcD + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0785  -0.7364  -0.2099  -0.0455   16.3720
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.83418    0.76566   7.620 1.65e-10 ***
## BAcDj45.1    0.35581    0.85069   0.418   0.677
## BAcDK        0.82539    0.66120   1.248   0.217
## Age         -0.02309    0.01965  -1.175   0.244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 4.977474)
##
## Null deviance: 327.87 on 66 degrees of freedom
## Residual deviance: 313.58 on 63 degrees of freedom
## (63 observations deleted due to missingness)
## AIC: 303.54
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL1b ~ BAcD + Age + Sex,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL1b ~ BAcD + Age + Sex, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.6520  -0.3432  -0.1722   0.0225   5.7759
##
## Coefficients:
```

```
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.655098   0.376076   7.060 1.69e-09 ***
## BAc dj45.1    0.365504   0.363494   1.006  0.3186
## BAc dK        0.270907   0.284396   0.953  0.3445
## Age          -0.014254   0.008466  -1.684  0.0973 .
## Sex1          -0.335396   0.257446  -1.303  0.1975
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.8835898)
##
## Null deviance: 59.026  on 66  degrees of freedom
## Residual deviance: 54.783  on 62  degrees of freedom
## (63 observations deleted due to missingness)
## AIC: 188.65
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL6 ~ BAc d + Age,
             family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL6 ~ BAc d + Age, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -49.986  -18.346  -10.042    5.672   157.568
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.9564    12.7374   2.823  0.00636 **
## BAc dj45.1    33.7804    14.1520   2.387  0.02000 *
## BAc dK        16.3414    10.9997   1.486  0.14236
## Age          -0.8424     0.3268  -2.577  0.01231 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1377.513)
##
## Null deviance: 99375  on 66  degrees of freedom
## Residual deviance: 86783  on 63  degrees of freedom
## (63 observations deleted due to missingness)
## AIC: 680.29
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL4RNA ~ BAc d,
             family = 'gaussian', data = bacit)
summary(m_cit)
```

```
##
## Call:
## glm(formula = IL4RNA ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.7410  -0.4468  -0.4468  -0.3622   5.4270
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.36215    0.29090   1.245   0.217
## BAcDj45.1     0.37883    0.42654   0.888   0.377
## BAcDK         0.08465    0.36507   0.232   0.817
##
## (Dispersion parameter for gaussian family taken to be 1.946283)
##
##      Null deviance: 157.41  on 82  degrees of freedom
## Residual deviance: 155.70  on 80  degrees of freedom
## (47 observations deleted due to missingness)
## AIC: 295.76
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(IL17RNA ~ BAcD,
              family = 'gaussian', data = bacit)
summary(m_cit)

##
## Call:
## glm(formula = IL17RNA ~ BAcD, family = "gaussian", data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -8.642  -2.122  -2.122  -0.296  164.163
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.2964    4.0547   0.073   0.942
## BAcDj45.1     8.3458    5.9454   1.404   0.164
## BAcDK         1.8261    5.0887   0.359   0.721
##
## (Dispersion parameter for gaussian family taken to be 378.1418)
##
##      Null deviance: 31084  on 82  degrees of freedom
## Residual deviance: 30251  on 80  degrees of freedom
## (47 observations deleted due to missingness)
## AIC: 733.12
##
## Number of Fisher Scoring iterations: 2

m_cit <- glm(TGFRNA ~ BAcD * TBLogAvr + Age,
              family = 'gaussian', data = bacit)
summary(m_cit)
```

```
##
## Call:
## glm(formula = TGFRNA ~ BAcD * TBLogAvr + Age, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -4.6512  -1.2514  -0.4214   0.4122  18.1527
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.081e+01  1.110e+01  -1.875   0.0661 .
## BAcDj45.1      1.875e+01  1.184e+01   1.584   0.1190
## BAcDK          1.937e+01  1.123e+01   1.726   0.0900 .
## TBLogAvr       6.956e+03  3.379e+03   2.059   0.0443 *
## Age           3.736e-02  3.460e-02   1.080   0.2851
## BAcDj45.1:TBLogAvr -6.775e+03  3.628e+03  -1.867   0.0672 .
## BAcDK:TBLogAvr   -6.574e+03  3.447e+03  -1.907   0.0617 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 12.71732)
##
##      Null deviance: 833.70  on 61  degrees of freedom
## Residual deviance: 699.45  on 55  degrees of freedom
## (68 observations deleted due to missingness)
## AIC: 342.18
##
## Number of Fisher Scoring iterations: 2

coef(m_cit)

##              (Intercept)              BAcDj45.1              BAcDK              TBLogAvr
##      -2.081035e+01      1.875090e+01      1.937461e+01      6.956495e+03
##              Age BAcDj45.1:TBLogAvr      BAcDK:TBLogAvr
##      3.735631e-02      -6.775400e+03      -6.573712e+03

##нет связи отдельных цитокинов с экспозицией
m_cit_PM <- glm(DPP4 ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = DPP4 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -787.23  -237.53   -3.53   194.63   646.66
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) 1278.665 231.315 5.528 5.95e-07 ***
## TBLogAvr -52693.281 52522.195 -1.003 0.3194
## Age 1.621 2.978 0.544 0.5880
## BMI -5.826 7.533 -0.773 0.4421
## Sex1 171.407 81.522 2.103 0.0393 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 96618.32)
##
## Null deviance: 6998263 on 70 degrees of freedom
## Residual deviance: 6376809 on 66 degrees of freedom
## (70 observations deleted due to missingness)
## AIC: 1023.3
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TSLP ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TSLP ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
## data = ba)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -52.12 -30.31 -16.37 -0.43 996.05
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 40.221 90.369 0.445 0.658
## TBLogAvr 4793.935 20062.426 0.239 0.812
## Age -1.435 1.128 -1.272 0.208
## BMI 1.156 2.916 0.397 0.693
## Sex1 -28.329 31.936 -0.887 0.378
##
## (Dispersion parameter for gaussian family taken to be 14985.3)
##
## Null deviance: 1081586 on 74 degrees of freedom
## Residual deviance: 1048971 on 70 degrees of freedom
## (66 observations deleted due to missingness)
## AIC: 940.78
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4 ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
```

```
## glm(formula = IL4 ~ TBlogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -0.9576   -0.5302   -0.2853    0.1034    5.3754
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.659356   0.751939   0.877   0.3836
## TBlogAvr     -21.467070  166.934339  -0.129   0.8980
## Age           -0.023373   0.009387  -2.490   0.0152 *
## BMI            0.034573   0.024259   1.425   0.1586
## Sex1          -0.158530   0.265728  -0.597   0.5527
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.037504)
##
##      Null deviance: 79.477  on 74  degrees of freedom
## Residual deviance: 72.625  on 70  degrees of freedom
## (66 observations deleted due to missingness)
## AIC: 222.43
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL5 ~ TBlogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL5 ~ TBlogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -8.652   -7.244   -4.733    1.202   68.044
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.28356   9.84645   0.029   0.977
## TBlogAvr     2761.42148 2185.96353   1.263   0.211
## Age           -0.06282   0.12292  -0.511   0.611
## BMI            0.09364   0.31767   0.295   0.769
## Sex1          -0.08477   3.47965  -0.024   0.981
##
## (Dispersion parameter for gaussian family taken to be 177.9034)
##
##      Null deviance: 12865  on 74  degrees of freedom
## Residual deviance: 12453  on 70  degrees of freedom
## (66 observations deleted due to missingness)
## AIC: 608.26
```

```
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL13 ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL13 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -76.68  -45.38  -24.56    3.66   457.25
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   52.6802    65.3861   0.806   0.423
## TBLogAvr    6753.1616 14516.0621   0.465   0.643
## Age           -0.1094     0.8163  -0.134   0.894
## BMI            0.3214     2.1095   0.152   0.879
## Sex1          -18.7024     23.1069  -0.809   0.421
##
## (Dispersion parameter for gaussian family taken to be 7845.058)
##
##      Null deviance: 556664  on 74  degrees of freedom
## Residual deviance: 549154  on 70  degrees of freedom
## (66 observations deleted due to missingness)
## AIC: 892.24
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL33 ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL33 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -131.38  -66.87  -40.73    9.33  1802.39
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   128.051    169.833   0.754   0.453
## TBLogAvr    11423.616  37703.744   0.303   0.763
## Age           -3.249     2.120  -1.533   0.130
## BMI            1.027     5.479   0.187   0.852
## Sex1          -54.160     60.017  -0.902   0.370
```

```
##
## (Dispersion parameter for gaussian family taken to be 52925.81)
##
## Null deviance: 3872846 on 74 degrees of freedom
## Residual deviance: 3704806 on 70 degrees of freedom
## (66 observations deleted due to missingness)
## AIC: 1035.4
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL25 ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL25 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
## data = ba)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -0.2288 -0.1278 -0.0753 -0.0027 3.9331
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.238948 0.359809 0.664 0.509
## TBLogAvr 19.078970 79.879571 0.239 0.812
## Age -0.005398 0.004492 -1.202 0.233
## BMI 0.001986 0.011608 0.171 0.865
## Sex1 -0.129457 0.127153 -1.018 0.312
##
## (Dispersion parameter for gaussian family taken to be 0.2375582)
##
## Null deviance: 17.165 on 74 degrees of freedom
## Residual deviance: 16.629 on 70 degrees of freedom
## (66 observations deleted due to missingness)
## AIC: 111.87
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFb1 ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFb1 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
## data = ba)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -63004 -7968 2376 15238 36911
##
```



```
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.873e+04  1.636e+04   3.591 0.000629 ***
## TBLogAvr    -4.655e+06  3.714e+06  -1.253 0.214466
## Age        -1.405e+02  2.106e+02  -0.667 0.506997
## BMI         1.007e+03  5.327e+02   1.891 0.063041 .
## Sex1       -4.553e+01  5.764e+03  -0.008 0.993722
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 483087749)
##
## Null deviance: 3.4110e+10 on 70 degrees of freedom
## Residual deviance: 3.1884e+10 on 66 degrees of freedom
## (70 observations deleted due to missingness)
## AIC: 1628
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17A ~ TBLogAvr + Age + BMI + Sex,
               family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17A ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0814  -0.7046  -0.2745   0.0146  16.3851
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.211774    1.623278   3.827 0.000291 ***
## TBLogAvr     91.506786  368.579501   0.248 0.804697
## Age         -0.027366    0.020898  -1.310 0.194903
## BMI          0.009703    0.052867   0.184 0.854946
## Sex1        -0.728039    0.572085  -1.273 0.207623
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 4.758128)
##
## Null deviance: 328.32 on 70 degrees of freedom
## Residual deviance: 314.04 on 66 degrees of freedom
## (70 observations deleted due to missingness)
## AIC: 319.05
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(IL1b ~ TBLLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL1b ~ TBLLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.5593  -0.3373  -0.1543  -0.0288   5.8227
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.29502    0.68200   3.365  0.00128 **
## TBLLogAvr    64.79597   154.85409   0.418  0.67699
## Age          -0.01368    0.00878  -1.558  0.12393
## BMI           0.01514    0.02221   0.681  0.49795
## Sex1         -0.37926    0.24035  -1.578  0.11936
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.8398836)
##
##      Null deviance: 59.165  on 70  degrees of freedom
## Residual deviance: 55.432  on 66  degrees of freedom
## (70 observations deleted due to missingness)
## AIC: 195.92
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL6 ~ TBLLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL6 ~ TBLLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -37.877  -19.998   -9.763   -0.835  167.151
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   29.9052    28.0423   1.066  0.2901
## TBLLogAvr    644.7557   6367.2471   0.101  0.9197
## Age          -0.8174    0.3610  -2.264  0.0269 *
## BMI           0.9033    0.9133   0.989  0.3262
## Sex1         -8.9207    9.8828  -0.903  0.3700
## ---

```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1419.963)
##
##    Null deviance: 101629  on 70  degrees of freedom
## Residual deviance:  93718  on 66  degrees of freedom
## (70 observations deleted due to missingness)
## AIC: 723.65
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4RNA ~ TBLogAvr + Age + BMI + Sex,
               family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4RNA ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.7610  -0.4782  -0.3488  -0.0702   5.5521
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.205430   0.869821   0.236   0.814
## TBLogAvr     181.133822 180.879169   1.001   0.321
## Age          -0.001861   0.011781  -0.158   0.875
## BMI           -0.014557   0.029204  -0.498   0.620
## Sex1          0.246773   0.309706   0.797   0.429
##
## (Dispersion parameter for gaussian family taken to be 1.38837)
##
##    Null deviance: 89.068  on 66  degrees of freedom
## Residual deviance: 86.079  on 62  degrees of freedom
## (74 observations deleted due to missingness)
## AIC: 218.93
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17RNA ~ TBLogAvr + Age + BMI + Sex,
               family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17RNA ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -4.2653  -2.1579  -1.0619  -0.0323  30.7755
```

```
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.26360    3.95704   1.077   0.2854
## TBLogAvr    -220.43119  822.86546  -0.268   0.7897
## Age           0.04917    0.05359   0.918   0.3624
## BMI          -0.19893    0.13285  -1.497   0.1394
## Sex1          2.52893    1.40893   1.795   0.0775 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 28.73332)
##
##      Null deviance: 1942.4  on 66  degrees of freedom
## Residual deviance: 1781.5  on 62  degrees of freedom
## (74 observations deleted due to missingness)
## AIC: 421.93
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFRNA ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFRNA ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.6119  -1.5172  -0.9543   0.2069  21.3814
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.374689    2.669794   0.515   0.608
## TBLogAvr     775.912413  555.183105   1.398   0.167
## Age           0.028979    0.036159   0.801   0.426
## BMI          -0.134407    0.089636  -1.499   0.139
## Sex1         -0.003078    0.950599  -0.003   0.997
##
## (Dispersion parameter for gaussian family taken to be 13.07979)
##
##      Null deviance: 861.75  on 66  degrees of freedom
## Residual deviance: 810.95  on 62  degrees of freedom
## (74 observations deleted due to missingness)
## AIC: 369.2
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(DPP4 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)
```

```
##
## Call:
## glm(formula = DPP4 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -771.79  -220.90   -11.02   194.08   646.05
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1267.416    216.117   5.864 1.52e-07 ***
## PM2.5Avr     -6321.484   5267.763  -1.200   0.2344
## Age           1.492      2.941    0.507   0.6136
## BMI          -4.954      7.279   -0.681   0.4985
## Sex1         175.163     80.656   2.172   0.0334 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 95183.89)
##
##      Null deviance: 7007453  on 71  degrees of freedom
## Residual deviance: 6377321  on 67  degrees of freedom
## (69 observations deleted due to missingness)
## AIC: 1036.5
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TSLP ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TSLP ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -50.07  -31.74  -14.87   -0.36   996.52
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   64.176    84.639   0.758   0.451
## PM2.5Avr     -299.848   2012.992  -0.149   0.882
## Age          -1.518     1.115  -1.362   0.178
## BMI           1.189     2.819   0.422   0.674
## Sex1         -28.146    31.610  -0.890   0.376
##
## (Dispersion parameter for gaussian family taken to be 14791.08)
##
##      Null deviance: 1081875  on 75  degrees of freedom
## Residual deviance: 1050167  on 71  degrees of freedom
```

```
## (65 observations deleted due to missingness)
## AIC: 952.24
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.9225  -0.5674  -0.2846   0.1013   5.4200
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.754208   0.706164   1.068   0.2891
## PM2.5Avr    -3.785875  16.794863  -0.225   0.8223
## Age         -0.023236   0.009302  -2.498   0.0148 *
## BMI          0.031215   0.023518   1.327   0.1887
## Sex1        -0.141825   0.263731  -0.538   0.5924
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.0296)
##
##      Null deviance: 79.757  on 75  degrees of freedom
## Residual deviance: 73.102  on 71  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 224.72
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL5 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL5 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -8.818  -6.897  -4.636   1.448  67.972
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.02035   9.23891   0.219   0.828
```

```
## PM2.5Avr    272.92459  219.73111   1.242   0.218
## Age         -0.06369   0.12171  -0.523   0.602
## BMI         0.07582   0.30769   0.246   0.806
## Sex1        -0.30925   3.45046  -0.090   0.929
##
## (Dispersion parameter for gaussian family taken to be 176.2377)
##
##      Null deviance: 12903  on 75  degrees of freedom
## Residual deviance: 12513  on 71  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 615.57
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL13 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL13 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -78.14  -45.35  -26.22   -0.34  456.40
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   55.86440    61.28947   0.911   0.365
## PM2.5Avr     810.14577   1457.66097   0.556   0.580
## Age          -0.08282     0.80737  -0.103   0.919
## BMI           0.10540     2.04116   0.052   0.959
## Sex1        -18.75020     22.88978  -0.819   0.415
##
## (Dispersion parameter for gaussian family taken to be 7755.839)
##
##      Null deviance: 558176  on 75  degrees of freedom
## Residual deviance: 550665  on 71  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 903.18
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL33 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL33 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
```

```
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -127.44   -63.81   -34.74    11.65   1804.21
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  172.521    159.129   1.084   0.282
## PM2.5Avr    -255.670    3784.594  -0.068   0.946
## Age          -3.401      2.096  -1.622   0.109
## BMI           1.078      5.300   0.203   0.839
## Sex1        -54.104     59.430  -0.910   0.366
##
## (Dispersion parameter for gaussian family taken to be 52282.24)
##
##      Null deviance: 3875372  on 75  degrees of freedom
## Residual deviance: 3712039  on 71  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 1048.2
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL25 ~ PM2.5Avr + Age + BMI + Sex,
               family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL25 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -0.2206   -0.1318   -0.0653    0.0033    3.9347
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.335872   0.336979   0.997   0.322
## PM2.5Avr    -1.269625   8.014457  -0.158   0.875
## Age         -0.005741   0.004439  -1.293   0.200
## BMI          0.002151   0.011223   0.192   0.849
## Sex1        -0.128774   0.125852  -1.023   0.310
##
## (Dispersion parameter for gaussian family taken to be 0.2344574)
##
##      Null deviance: 17.171  on 75  degrees of freedom
## Residual deviance: 16.646  on 71  degrees of freedom
## (65 observations deleted due to missingness)
## AIC: 112.27
##
## Number of Fisher Scoring iterations: 2
```



```

m_cit_PM <- glm(TGFb1 ~ PM2.5Avr + Age + BMI + Sex,
                 family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFb1 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -62618   -8586    1892   15288   37009
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  54486.32   15692.34   3.472 0.000909 ***
## PM2.5Avr    -250094.19  382494.20  -0.654 0.515446
## Age          -97.29     213.55  -0.456 0.650166
## BMI           771.80     528.55   1.460 0.148903
## Sex1         1112.19     5856.46   0.190 0.849955
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 501834709)
##
##      Null deviance: 3.4852e+10  on 71  degrees of freedom
## Residual deviance: 3.3623e+10  on 67  degrees of freedom
## (69 observations deleted due to missingness)
## AIC: 1653.6
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17A ~ PM2.5Avr + Age + BMI + Sex,
                 family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17A ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0397  -0.6846  -0.2484   0.0197  16.3843
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.80051    1.51659   4.484 2.94e-05 ***
## PM2.5Avr     -11.56466    36.96627  -0.313   0.755
## Age          -0.02961     0.02064  -1.434   0.156
## BMI           0.01198     0.05108   0.234   0.815
## Sex1         -0.72040     0.56600  -1.273   0.207
## ---

```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 4.687294)
##
##    Null deviance: 328.43  on 71  degrees of freedom
## Residual deviance: 314.05  on 67  degrees of freedom
## (69 observations deleted due to missingness)
## AIC: 322.37
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL1b ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL1b ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.5338  -0.3501  -0.1346  -0.0297   5.8378
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.510550   0.638660   3.931 0.000203 ***
## PM2.5Avr     0.003899  15.567076   0.000 0.999801
## Age          -0.014347   0.008691  -1.651 0.103464
## BMI           0.015053   0.021511   0.700 0.486504
## Sex1         -0.377759   0.238351  -1.585 0.117702
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.8312374)
##
##    Null deviance: 59.197  on 71  degrees of freedom
## Residual deviance: 55.693  on 67  degrees of freedom
## (69 observations deleted due to missingness)
## AIC: 197.84
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL6 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL6 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
```

```
##      Min      1Q   Median      3Q      Max
## -36.679 -20.464  -9.791   0.589  168.705
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   34.9505    26.2840   1.330   0.1881
## PM2.5Avr     -43.8527   640.6625  -0.068   0.9456
## Age          -0.8205     0.3577  -2.294   0.0249 *
## BMI           0.8069     0.8853   0.911   0.3654
## Sex1         -8.4751     9.8093  -0.864   0.3907
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1407.893)
##
##      Null deviance: 102045  on 71  degrees of freedom
## Residual deviance:  94329  on 67  degrees of freedom
## (69 observations deleted due to missingness)
## AIC: 733.13
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4RNA ~ PM2.5Avr + Age + BMI + Sex,
               family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4RNA ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.8158  -0.4975  -0.3438  -0.0025   5.5220
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.105681   0.817791   0.129   0.898
## PM2.5Avr     25.719558  18.622279   1.381   0.172
## Age          -0.001536   0.011558  -0.133   0.895
## BMI          -0.014800   0.028112  -0.526   0.600
## Sex1         0.224014   0.303129   0.739   0.463
##
## (Dispersion parameter for gaussian family taken to be 1.34797)
##
##      Null deviance: 89.210  on 67  degrees of freedom
## Residual deviance: 84.922  on 63  degrees of freedom
## (73 observations deleted due to missingness)
## AIC: 220.09
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(IL17RNA ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17RNA ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -4.3709  -2.1688  -0.9860  -0.0543   30.7219
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.78543    3.74846   1.010   0.3164
## PM2.5Avr    -10.37657   85.35775  -0.122   0.9036
## Age           0.05024    0.05298   0.948   0.3466
## BMI          -0.19607    0.12886  -1.522   0.1331
## Sex1          2.53955    1.38943   1.828   0.0723 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 28.32048)
##
##      Null deviance: 1944.2  on 67  degrees of freedom
## Residual deviance: 1784.2  on 63  degrees of freedom
## (73 observations deleted due to missingness)
## AIC: 427.15
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFRNA ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFRNA ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.3626  -1.5414  -0.9830  -0.1915   21.1550
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.10057    2.67800   0.411   0.682
## PM2.5Avr     67.44268   60.98193   1.106   0.273
## Age           0.02326    0.03785   0.614   0.541
## BMI          -0.08100    0.09206  -0.880   0.382
## Sex1         -0.31075    0.99265  -0.313   0.755
##

```

```
## (Dispersion parameter for gaussian family taken to be 14.45496)
##
## Null deviance: 940.11 on 67 degrees of freedom
## Residual deviance: 910.66 on 63 degrees of freedom
## (73 observations deleted due to missingness)
## AIC: 381.41
##
## Number of Fisher Scoring iterations: 2

#TSLP, IL33, IL25, ИЛ17, ИЛ16ета - обратная связь с ЛАЛ1
summary(ba$LAL1)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 0.00640 0.01390 0.02760 0.02243 0.02770 0.02770 67

#ИЛ4рнк, TGFбрнк - прямая связь с ЛАЛ2 (тенденция)
summary(ba$LAL2)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## 0.02790 0.03820 0.05530 0.05452 0.06940 0.06940 67

m_cit_PM <- glm(DPP4 ~ LAL1 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = DPP4 ~ LAL1 + Age + BMI + Sex, family = "gaussian",
## data = ba)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -807.90 -240.52 14.48 188.87 623.23
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1150.885 215.537 5.340 1.83e-06 ***
## LAL1 -1395.713 5247.208 -0.266 0.7912
## Age 2.039 3.283 0.621 0.5372
## BMI -6.516 8.066 -0.808 0.4227
## Sex1 182.694 94.443 1.934 0.0582 .
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 104339.7)
##
## Null deviance: 6206867 on 59 degrees of freedom
## Residual deviance: 5738686 on 55 degrees of freedom
## (81 observations deleted due to missingness)
## AIC: 870.38
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(TSLP ~ LAL1 + Age,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TSLP ~ LAL1 + Age, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -109.38   -24.02   -11.02    10.50   934.93
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   166.562     63.273   2.632  0.0107 *
## LAL1          -4119.175    1985.491  -2.075  0.0422 *
## Age            -1.246       1.082  -1.151  0.2543
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 16183.31)
##
##      Null deviance: 1077699  on 63  degrees of freedom
## Residual deviance:  987182  on 61  degrees of freedom
## (77 observations deleted due to missingness)
## AIC: 806.82
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4 ~ LAL1 + Age + BMI,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4 ~ LAL1 + Age + BMI, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
##  -0.9923  -0.4831  -0.2633   0.0911   5.3928
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.636812   0.686466   0.928  0.3574
## LAL1          -7.267891   16.365877  -0.444  0.6586
## Age           -0.024997   0.009895  -2.526  0.0142 *
## BMI            0.040792   0.025562   1.596  0.1159
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.087384)
##
##      Null deviance: 71.565  on 62  degrees of freedom

```

```
## Residual deviance: 64.156 on 59 degrees of freedom
## (78 observations deleted due to missingness)
## AIC: 189.93
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL5 ~ LAL1,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL5 ~ LAL1, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -8.028  -7.519  -4.901   0.455  68.732
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    6.431     5.458   1.178   0.243
## LAL1          123.354    220.896   0.558   0.579
##
## (Dispersion parameter for gaussian family taken to be 200.3227)
##
##      Null deviance: 12482 on 63 degrees of freedom
## Residual deviance: 12420 on 62 degrees of freedom
## (77 observations deleted due to missingness)
## AIC: 524.79
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL13 ~ LAL1,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL13 ~ LAL1, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -73.34  -48.24  -27.12   8.94  460.26
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    48.97     33.99   1.440   0.155
## LAL1          994.82    1375.84   0.723   0.472
##
## (Dispersion parameter for gaussian family taken to be 7771.317)
##
##      Null deviance: 485885 on 63 degrees of freedom
## Residual deviance: 481822 on 62 degrees of freedom
```

```
## (77 observations deleted due to missingness)
## AIC: 758.92
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL33 ~ LAL1 + Age,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL33 ~ LAL1 + Age, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -235.83   -71.69   -27.92    22.20   1673.27
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   372.845    117.795   3.165  0.00242 **
## LAL1          -8298.079    3696.336  -2.245  0.02841 *
## Age            -3.277      2.015  -1.627  0.10899
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 56088.59)
##
##      Null deviance: 3849424  on 63  degrees of freedom
## Residual deviance: 3421404  on 61  degrees of freedom
## (77 observations deleted due to missingness)
## AIC: 886.37
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL25 ~ LAL1 + Age,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL25 ~ LAL1 + Age, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
##  -0.4759   -0.1115   -0.0477    0.0380    3.6798
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.708195    0.250839   2.823  0.00641 **
## LAL1          -16.987004    7.871201  -2.158  0.03486 *
## Age            -0.005284    0.004290  -1.232  0.22285
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



```
##
## (Dispersion parameter for gaussian family taken to be 0.2543394)
##
## Null deviance: 17.075 on 63 degrees of freedom
## Residual deviance: 15.515 on 61 degrees of freedom
## (77 observations deleted due to missingness)
## AIC: 98.93
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFb1 ~ LAL1 + BMI,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFb1 ~ LAL1 + BMI, family = "gaussian", data = ba)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -63154 -8333 1907 16606 35042
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 33034.2 14780.8 2.235 0.0294 *
## LAL1 423604.3 359334.0 1.179 0.2434
## BMI 824.7 508.6 1.621 0.1104
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 517882685)
##
## Null deviance: 3.1754e+10 on 59 degrees of freedom
## Residual deviance: 2.9519e+10 on 57 degrees of freedom
## (81 observations deleted due to missingness)
## AIC: 1379.1
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17A ~ LAL1 + Age,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17A ~ LAL1 + Age, family = "gaussian", data = ba)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -2.4617 -0.3675 -0.2103 0.2959 14.9680
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
```

```

## (Intercept)    8.73965    1.10740    7.892 9.29e-11 ***
## LAL1          -97.62680   34.57958   -2.823  0.0065 **
## Age           -0.02537    0.01912   -1.327  0.1898
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 4.842439)
##
##      Null deviance: 327.07  on 60  degrees of freedom
## Residual deviance: 280.86  on 58  degrees of freedom
## (80 observations deleted due to missingness)
## AIC: 274.26
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL1b ~ LAL1 + Age,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL1b ~ LAL1 + Age, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.9742  -0.2331  -0.1667   0.0530   5.3825
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.457720   0.479281   7.214 1.28e-09 ***
## LAL1          -34.502926  14.965907  -2.305  0.0247 *
## Age           -0.010602   0.008275  -1.281  0.2052
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.9070492)
##
##      Null deviance: 58.784  on 60  degrees of freedom
## Residual deviance: 52.609  on 58  degrees of freedom
## (80 observations deleted due to missingness)
## AIC: 172.08
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL6 ~ LAL1 + Age,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL6 ~ LAL1 + Age, family = "gaussian", data = ba)
##
## Deviance Residuals:

```

```
##      Min      1Q   Median      3Q      Max
## -34.769 -20.230 -10.332  -3.305  175.723
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   53.2177    19.6797   2.704  0.00897 **
## LAL1         -341.2411   614.5147  -0.555  0.58082
## Age           -0.6112     0.3398  -1.799  0.07724 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1529.288)
##
##      Null deviance: 94043  on 60  degrees of freedom
## Residual deviance: 88699  on 58  degrees of freedom
## (80 observations deleted due to missingness)
## AIC: 625.32
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4RNA ~ LAL1,
               family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4RNA ~ LAL1, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.6168  -0.6136  -0.5966   0.0535   5.2570
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.2549     0.4297  -0.593  0.5554
## LAL1          31.4690    18.4485   1.706  0.0938 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.51448)
##
##      Null deviance: 86.189  on 55  degrees of freedom
## Residual deviance: 81.782  on 54  degrees of freedom
## (85 observations deleted due to missingness)
## AIC: 186.13
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17RNA ~ LAL1 + Age + BMI + Sex,
               family = 'gaussian', data = ba)
summary(m_cit_PM)
```

```
##
## Call:
## glm(formula = IL17RNA ~ LAL1 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -5.1137  -2.5497  -1.2608  -0.1939   30.1569
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.54664    4.10528   1.108   0.2734
## LAL1          -12.86852   91.65163  -0.140   0.8889
## Age            0.05845    0.06486   0.901   0.3718
## BMI           -0.23285    0.15241  -1.528   0.1329
## Sex1           2.93489    1.74357   1.683   0.0986 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 34.73816)
##
##      Null deviance: 1917.2  on 54  degrees of freedom
## Residual deviance: 1736.9  on 50  degrees of freedom
## (86 observations deleted due to missingness)
## AIC: 357.97
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFRNA ~ LAL1,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFRNA ~ LAL1, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.6609  -1.6306  -0.9531  -0.2496   21.8277
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.2133    1.3464   0.158   0.875
## LAL1          52.2596   57.8106   0.904   0.370
##
## (Dispersion parameter for gaussian family taken to be 14.87157)
##
##      Null deviance: 815.22  on 55  degrees of freedom
## Residual deviance: 803.06  on 54  degrees of freedom
## (85 observations deleted due to missingness)
## AIC: 314.05
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(DPP4 ~ LAL2 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = DPP4 ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -768.25  -252.09   20.68   197.64   623.55
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1276.396    250.363   5.098 4.36e-06 ***
## LAL2         -2914.876    3116.532  -0.935  0.3537
## Age           1.604       3.294   0.487  0.6282
## BMI          -5.594       8.061  -0.694  0.4906
## Sex1         182.545     91.606   1.993  0.0513 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 102838.3)
##
##      Null deviance: 6206867  on 59  degrees of freedom
## Residual deviance: 5656107  on 55  degrees of freedom
## (81 observations deleted due to missingness)
## AIC: 869.51
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TSLP ~ LAL2 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TSLP ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -92.73  -34.22  -13.52    9.85   952.91
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   162.781    102.652   1.586  0.118
## LAL2         -1815.445    1267.170  -1.433  0.157
## Age           -2.027     1.313  -1.544  0.128
## BMI           1.965     3.251   0.604  0.548
## Sex1         -29.905    37.335  -0.801  0.426
##

```

```
## (Dispersion parameter for gaussian family taken to be 17333.81)
##
## Null deviance: 1077552 on 62 degrees of freedom
## Residual deviance: 1005361 on 58 degrees of freedom
## (78 observations deleted due to missingness)
## AIC: 800.48
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4 ~ LAL2 + Age + BMI,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4 ~ LAL2 + Age + BMI, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.9951  -0.4848  -0.2381   0.0743   5.3926
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.61824    0.79910   0.774  0.4422
## LAL2        -2.55564    10.03511  -0.255  0.7999
## Age         -0.02519     0.01003  -2.513  0.0147 *
## BMI          0.04084     0.02578   1.584  0.1185
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.08982)
##
## Null deviance: 71.565 on 62 degrees of freedom
## Residual deviance: 64.299 on 59 degrees of freedom
## (78 observations deleted due to missingness)
## AIC: 190.07
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL5 ~ LAL2 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL5 ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -11.125   -7.403   -4.793    0.748   66.029
##
## Coefficients:
```

```

##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.0829    11.2029   0.097   0.923
## LAL2          167.4226   138.2920   1.211   0.231
## Age           -0.1077    0.1433  -0.752   0.455
## BMI            0.1160    0.3548   0.327   0.745
## Sex1          -0.8103    4.0745  -0.199   0.843
##
## (Dispersion parameter for gaussian family taken to be 206.4517)
##
##      Null deviance: 12463  on 62  degrees of freedom
## Residual deviance: 11974  on 58  degrees of freedom
## (78 observations deleted due to missingness)
## AIC: 521.37
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL13 ~ LAL2 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL13 ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -90.20  -47.10  -24.37    7.85   450.19
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  47.9681    70.3452   0.682   0.498
## LAL2         648.5389   868.3634   0.747   0.458
## Age          -0.5542    0.8995  -0.616   0.540
## BMI           0.6021    2.2280   0.270   0.788
## Sex1        -23.6019   25.5847  -0.922   0.360
##
## (Dispersion parameter for gaussian family taken to be 8140.059)
##
##      Null deviance: 485325  on 62  degrees of freedom
## Residual deviance: 472123  on 58  degrees of freedom
## (78 observations deleted due to missingness)
## AIC: 752.86
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL33 ~ LAL2 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL33 ~ LAL2 + Age + BMI + Sex, family = "gaussian",

```

```
##      data = ba)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -196.13   -68.02   -31.01    12.84   1710.35
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   391.236    191.485   2.043  0.0456 *
## LAL2          -3769.344   2363.757  -1.595  0.1162
## Age            -4.661      2.449  -1.904  0.0619 .
## BMI             2.823      6.065   0.465  0.6434
## Sex1          -56.734     69.644  -0.815  0.4186
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 60315.7)
##
##      Null deviance: 3848648  on 62  degrees of freedom
## Residual deviance: 3498311  on 58  degrees of freedom
## (78 observations deleted due to missingness)
## AIC: 879.04
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL25 ~ LAL2 + Age + BMI + Sex,
               family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL25 ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -0.3999   -0.1397   -0.0642    0.0267    3.7539
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.743149   0.407317   1.824  0.0732 .
## LAL2          -7.331350   5.028056  -1.458  0.1502
## Age           -0.008178   0.005208  -1.570  0.1218
## BMI            0.005406   0.012901   0.419  0.6767
## Sex1          -0.138938   0.148142  -0.938  0.3522
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.2729133)
##
##      Null deviance: 17.070  on 62  degrees of freedom
## Residual deviance: 15.829  on 58  degrees of freedom
## (78 observations deleted due to missingness)
```



```
## AIC: 103.76
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFb1 ~ LAL2 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFb1 ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -62093   -8609    2608   14876   33285
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  27762.0     17936.9   1.548   0.127
## LAL2         288656.3    223279.9   1.293   0.201
## Age          -123.5       236.0  -0.523   0.603
## BMI           950.3       577.5   1.646   0.106
## Sex1          218.1       6563.0   0.033   0.974
##
## (Dispersion parameter for gaussian family taken to be 527850715)
##
##      Null deviance: 3.1754e+10  on 59  degrees of freedom
## Residual deviance: 2.9032e+10  on 55  degrees of freedom
## (81 observations deleted due to missingness)
## AIC: 1382.1
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17A ~ LAL2 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17A ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0785  -0.6179  -0.2951   0.1784  15.4051
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.97589    1.79721   4.994 6.32e-06 ***
## LAL2          -42.11771    22.37175  -1.883   0.065 .
## Age           -0.04018     0.02365  -1.699   0.095 .
## BMI            0.02732     0.05786   0.472   0.639
```

```
## Sex1          -0.77952    0.65759  -1.185    0.241
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 5.299215)
##
##      Null deviance: 326.92  on 59  degrees of freedom
## Residual deviance: 291.46  on 55  degrees of freedom
## (81 observations deleted due to missingness)
## AIC: 277.11
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL1b ~ LAL2 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL1b ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.8114  -0.3537  -0.1428   0.0375   5.4962
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.28617    0.76521   4.294 7.18e-05 ***
## LAL2          -12.78202    9.52537  -1.342  0.1851
## Age           -0.01897    0.01007  -1.884  0.0649 .
## BMI            0.02177    0.02464   0.884  0.3807
## Sex1          -0.42111    0.27998  -1.504  0.1383
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.9606727)
##
##      Null deviance: 58.739  on 59  degrees of freedom
## Residual deviance: 52.837  on 55  degrees of freedom
## (81 observations deleted due to missingness)
## AIC: 174.64
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL6 ~ LAL2 + Age + BMI + Sex,
                family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL6 ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
```

```
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -40.144  -20.191  -10.850   -1.267   165.836
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   40.3376    30.6168   1.318   0.1931
## LAL2          -129.2607   381.1190  -0.339   0.7358
## Age            -0.9623     0.4029  -2.389   0.0204 *
## BMI            1.1356     0.9857   1.152   0.2543
## Sex1          -10.7027    11.2025  -0.955   0.3436
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1537.917)
##
##      Null deviance: 93622  on 59  degrees of freedom
## Residual deviance: 84585  on 55  degrees of freedom
## (81 observations deleted due to missingness)
## AIC: 617.34
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4RNA ~ LAL2,
               family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4RNA ~ LAL2, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.7372  -0.6320  -0.4423  -0.1278   5.3506
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.6183     0.6312  -0.980   0.3317
## LAL2          19.5320    11.4395   1.707   0.0935 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.51433)
##
##      Null deviance: 86.189  on 55  degrees of freedom
## Residual deviance: 81.774  on 54  degrees of freedom
## (85 observations deleted due to missingness)
## AIC: 186.12
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(IL17RNA ~ LAL2 + Age + BMI + Sex,
                 family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17RNA ~ LAL2 + Age + BMI + Sex, family = "gaussian",
##      data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -5.319  -2.576  -1.272  -0.088   29.986
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.84677    4.82263   0.798   0.4288
## LAL2           8.62770   56.05833   0.154   0.8783
## Age            0.05953    0.06543   0.910   0.3673
## BMI           -0.23470    0.15270  -1.537   0.1306
## Sex1           2.86458    1.68967   1.695   0.0962 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 34.7354)
##
##      Null deviance: 1917.2  on 54  degrees of freedom
## Residual deviance: 1736.8  on 50  degrees of freedom
## (86 observations deleted due to missingness)
## AIC: 357.97
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFRNA ~ LAL2,
                 family = 'gaussian', data = ba)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFRNA ~ LAL2, family = "gaussian", data = ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.2962  -1.4942  -1.0576   0.1697  21.1924
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.827    1.942   -0.941   0.3510
## LAL2          59.418   35.202   1.688   0.0972 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 14.34003)
##

```

```
## Null deviance: 815.22 on 55 degrees of freedom
## Residual deviance: 774.36 on 54 degrees of freedom
## (85 observations deleted due to missingness)
## AIC: 312.02
##
## Number of Fisher Scoring iterations: 2
```

#Неаллергический фенотип

```
bacit <- ba[ba$BAcd %in% c("j45.1"), ]
str(bacit)
```

```
## 'data.frame': 42 obs. of 106 variables:
## $ ID : chr "60" "74" "71" "58" ...
## $ Kazan : Factor w/ 2 levels "0","1": 1 2 1 2 1 2 2 1 1 2 ...
## $ Point : Factor w/ 10 levels "1","4","5","6",...: NA 1 NA 1 NA 1 2
NA NA 1 ...
## $ TSPAвр : num NA 0.164 NA 0.164 NA 0.164 0.09 NA NA 0.164 ...
## $ TSPMaxAвр : num NA 0.595 NA 0.595 NA ...
## $ TSPMaxMax : num NA 0.734 NA 0.734 NA 0.734 0.437 NA NA 0.734 ...
## $ PM10Aвр : num NA 0.094 NA 0.094 NA 0.094 0.036 NA NA 0.094 ...
## $ PM10MaxAвр : num NA 0.348 NA 0.348 NA ...
## $ PM10MaxMax : num NA 0.44 NA 0.44 NA 0.44 0.285 NA NA 0.44 ...
## $ PM2.5Aвр : num NA 0.031 NA 0.031 NA 0.031 0.009 NA NA 0.031 ...
## $ PM2.5MaxAвр : num NA 0.172 NA 0.172 NA 0.172 0.0592 NA NA 0.172 ...
## $ PM2.5MaxMax : num NA 0.253 NA 0.253 NA 0.253 0.157 NA NA 0.253 ...
## $ TSPDT : num NA 0.038 NA 0.038 NA 0.038 0.026 NA NA 0.038 ...
## $ PM10DT : num NA 0.03 NA 0.03 NA 0.03 0.02 NA NA 0.03 ...
## $ PM25DT : num NA 0.028 NA 0.028 NA 0.028 0.018 NA NA 0.028 ...
## $ Cu : num NA 0.56 NA 0.56 NA 0.56 NA NA NA 0.56 ...
## $ C : num NA 93.4 NA 93.4 NA ...
## $ LAL1 : num NA 0.0277 NA 0.0277 NA 0.0277 NA NA NA 0.0277 ...
## $ LAL2 : num NA 0.0694 NA 0.0694 NA 0.0694 NA NA NA 0.0694 ...
## $ TBPLogMax : num NA 0.0221 NA 0.0221 NA ...
## $ PLogMax : num NA 0.0126 NA 0.0126 NA ...
## $ TBLogMax : num NA 0.00948 NA 0.00948 NA ...
## $ TBPLogAвр : num NA 0.00804 NA 0.00804 NA ...
## $ PLogAвр : num NA 0.00457 NA 0.00457 NA ...
## $ TBLogAвр : num NA 0.00346 NA 0.00346 NA ...
## $ BA : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
## $ BAcd : Factor w/ 4 levels "j45.0","j45.1",...: 2 2 2 2 2 2 2 2 2 2
2 ...
## $ BMI : num 39 39.1 22.7 27.9 26.8 ...
## $ BMIcd : Factor w/ 3 levels "1","2","3": 3 3 1 2 2 3 2 1 1 1 ...
## $ BAsevere : Factor w/ 3 levels "1","2","3": 3 2 3 2 2 1 2 2 3 3 ...
## $ BAobstr : Factor w/ 2 levels "0","1": 1 2 2 1 1 1 1 1 2 2 ...
## $ BAcontrol : Factor w/ 4 levels "1","2","3","4": 2 2 NA 3 2 NA 1 2 1
NA ...
## $ BAdebut : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 2 1 1 1 ...
## $ IgE : num 72.9 11.3 NA NA NA ...
## $ Eos : num 291.2 286 32.4 370 54 ...
## $ Atopia : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 2 1 ...
## $ Heredity : Factor w/ 2 levels "0","1": 1 2 1 1 2 1 1 1 1 2 ...
## $ Age : num 56.7 37.7 58.7 54.7 39.9 ...
```

```

## $ Sex          : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 1 2 1 2 ...
## $ Family       : Factor w/ 6 levels "1","2","3","4",...: 6 6 6 2 6 1 5 6 4
1 ...
## $ FamType      : Factor w/ 5 levels "0","1","2","3",...: 4 4 4 3 4 4 4 4 3
1 ...
## $ Child        : Factor w/ 6 levels "0","1","2","3",...: 3 2 2 3 4 1 1 2 2
1 ...
## $ Child_8      : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 1 1 1 ...
## $ Hobby        : Factor w/ 2 levels "0","1": 1 2 2 1 2 1 1 1 1 1 ...
## $ Educ         : Factor w/ 5 levels "2","3","4","5",...: 4 4 3 2 2 4 4 4 4
5 ...
## $ EducYrs      : num 15 16 13 13 13 16 15 17 19 15 ...
## $ ScaleS       : Factor w/ 9 levels "1","3","4","5",...: 7 6 6 3 4 3 5 2 8
5 ...
## $ ScaleR       : Factor w/ 10 levels "1","2","3","4",...: 5 3 7 4 5 5 4 3 8
6 ...
## $ ISL_cd       : Factor w/ 4 levels "1","2","3","4": 3 2 2 3 3 3 2 2 3 3
...
## $ ProfCateg    : Factor w/ 5 levels "1","2","3","4",...: 2 NA 2 4 NA 2 5 NA
NA 2 ...
## $ DustCurrent  : Factor w/ 2 levels "0","1": NA 1 1 1 2 1 NA 1 1 1 ...
## $ pDustAn      : int 1 NA 2 NA NA 0 5 5 0 0 ...
## $ Dust         : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 2 2 1 1 ...
## $ ChemCurrent  : Factor w/ 2 levels "0","1": NA 1 2 2 2 1 NA 2 1 2 ...
## $ Chem         : Factor w/ 2 levels "0","1": 2 1 2 2 2 1 2 2 1 2 ...
## $ BioCurrent   : Factor w/ 2 levels "0","1": NA 2 1 1 2 1 NA 1 1 1 ...
## $ Bio         : Factor w/ 2 levels "0","1": 1 2 1 1 2 1 2 1 1 1 ...
## $ ColdCurrent  : Factor w/ 2 levels "0","1": NA 2 2 2 2 1 NA 2 1 2 ...
## $ Cold        : Factor w/ 2 levels "0","1": 1 2 2 2 2 1 2 2 1 2 ...
## $ HeatCurrent  : Factor w/ 2 levels "0","1": NA 1 1 2 2 1 NA 2 1 1 ...
## $ Heat        : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 1 2 1 2 ...
## $ PhysLoadCurrent: Factor w/ 2 levels "0","1": NA 2 2 2 2 1 NA 2 1 NA ...
## $ PhysLoad     : Factor w/ 2 levels "0","1": 1 2 2 2 2 1 2 2 1 2 ...
## $ DL           : int NA 60 66 78 60 76 NA 58 NA 88 ...
## $ JD           : int NA 23 21 23 22 23 NA 32 NA 25 ...
## $ CS           : int NA 17 18 18 16 12 NA 17 NA 14 ...
## $ SS           : int NA NA 17 18 15 14 NA 7 NA 12 ...
## $ SocSup       : int NA NA 35 36 31 26 NA 24 NA 26 ...
## $ JobDis       : num NA 0.1 0 0.1 0 ...
## $ JI           : int NA 10 12 12 11 16 NA 15 NA 15 ...
## $ JDJI        : int NA 33 33 35 33 39 NA 47 NA 40 ...
## $ PE          : int NA 11 10 13 12 8 NA 9 NA 8 ...
## $ JCQedcd     : Factor w/ 2 levels "0","1": NA NA 1 1 1 1 NA 1 NA 1 ...
## $ ERI_Ef_Ph   : int NA 10 7 17 10 17 NA 20 NA NA ...
## $ ERI_Ef_M    : int NA 9 6 14 9 16 NA 19 NA NA ...
## $ Overcom     : int NA 12 13 18 12 16 NA 18 NA NA ...
## $ SANx        : int 46 42 29 52 NA 42 52 NA 46 42 ...
## $ SANx_cd     : Factor w/ 3 levels "1","2","3": 3 2 1 3 NA 2 3 NA 3 2 ...
## $ LE          : int 53 71 133 238 178 265 71 237 144 25 ...
## $ LE_cd       : Factor w/ 3 levels "1","2","3": 1 1 1 2 1 2 1 2 1 1 ...
## $ FamilyPast  : Factor w/ 2 levels "2","3": 2 2 2 2 2 2 2 2 1 NA ...
## $ SMK         : Factor w/ 3 levels "0","1","2": 1 2 1 1 1 1 1 2 NA NA ...

```

```
## $ PasSMK      : int  0 3 0 2 0 0 0 2 NA NA ...
## $ PARigMin    : int  240 360 40 60 NA NA 300 0 NA NA ...
## $ PAModMin    : int  120 420 30 60 NA NA 120 0 NA NA ...
## $ WalkMin     : int  120 1800 180 60 120 NA 240 50 NA NA ...
## $ FruVegs     : int  0 0 0 0 0 1 0 0 NA NA ...
## $ Omega3      : Factor w/ 2 levels "0","1": 1 1 2 2 2 2 1 1 NA NA ...
## $ DPP4        : num  NA 926 NA 1069 NA ...
## $ TGFb1       : num  NA 76425 NA 99988 NA ...
## $ TSLP        : num  NA 4.62 NA 4.62 NA ...
## $ IL13        : num  NA 102.3 NA 40.7 NA ...
## $ IL17A       : num  NA 5.18 NA 5.18 NA 5.18 5.18 NA NA 5.18 ...
## $ IL1b        : num  NA 2.06 NA 2.06 NA ...
## $ IL33        : num  NA 5.65 NA 1.79 NA ...
## $ IL4         : num  NA 1.31 NA 0.005 NA 6.21 0.005 NA NA 0.005 ...
## $ IL5         : num  NA 22.51 NA 2.32 NA ...
## $ IL6         : num  NA 46 NA 1.4 NA ...
## $ IL25        : num  NA 0.025 NA 0.025 NA 0.025 0.025 NA NA 0.025 ...
## [list output truncated]
```

```
m_cit_PM <- glm(DPP4 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)
```

```
##
## Call:
## glm(formula = DPP4 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -348.91  -252.72   -17.65   172.94   712.29
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   770.0138    584.5317   1.317   0.2203
## PM2.5Avr     1004.0251  14180.8551   0.071   0.9451
## Age           -0.8049     6.9081  -0.117   0.9098
## BMI            4.3024    14.1453   0.304   0.7679
## Sex1          475.4218    253.7598   1.874   0.0938 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 121786.3)
##
##      Null deviance: 1613202  on 13  degrees of freedom
## Residual deviance: 1096076  on  9  degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 209.48
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(TSLP ~ PM2.5Avr + Age + BMI + Sex,
                 family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TSLP ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -14.450   -7.853   -4.055    0.114   64.804
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.9595     35.9823   0.277   0.787
## PM2.5Avr    377.8166    849.0666   0.445   0.665
## Age          -0.3590     0.4081  -0.880   0.398
## BMI           0.2625     0.8199   0.320   0.755
## Sex1         -9.0879    15.0530  -0.604   0.558
##
## (Dispersion parameter for gaussian family taken to be 462.6308)
##
##      Null deviance: 5750.5  on 15  degrees of freedom
## Residual deviance: 5088.9  on 11  degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 149.6
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4 ~ PM2.5Avr + Age + BMI + Sex,
                 family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3730   -0.7344   -0.1956    0.2036    4.5875
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.49385     2.64542  -0.187   0.855
## PM2.5Avr    45.44978    62.42336   0.728   0.482
## Age         -0.03552     0.03000  -1.184   0.261
## BMI          0.06345     0.06028   1.053   0.315
## Sex1        -0.36402     1.10669  -0.329   0.748
##
## (Dispersion parameter for gaussian family taken to be 2.500606)
##

```



```
## Null deviance: 36.338 on 15 degrees of freedom
## Residual deviance: 27.507 on 11 degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 66.075
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL5 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL5 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -21.913   -9.995   -2.790    0.776   56.242
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -9.3161    34.6459  -0.269   0.793
## PM2.5Avr     695.4506   817.5327   0.851   0.413
## Age          -0.3321     0.3929  -0.845   0.416
## BMI           0.7090     0.7895   0.898   0.388
## Sex1          8.5073    14.4939   0.587   0.569
##
## (Dispersion parameter for gaussian family taken to be 428.9052)
##
## Null deviance: 6155.4 on 15 degrees of freedom
## Residual deviance: 4718.0 on 11 degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 148.39
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL13 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL13 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -103.22   -58.96   -30.84    40.27   195.57
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.9165    180.2436   0.022   0.983
```

```
## PM2.5Avr      5774.5167  4253.1740   1.358   0.202
## Age           -0.7201    2.0441  -0.352   0.731
## BMI           -0.5389    4.1071  -0.131   0.898
## Sex1          -72.9412   75.4038  -0.967   0.354
##
## (Dispersion parameter for gaussian family taken to be 11608.53)
##
##      Null deviance: 156868  on 15  degrees of freedom
## Residual deviance: 127694  on 11  degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 201.16
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL33 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL33 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -87.59  -48.75  -15.81   20.38  233.82
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  242.400    151.774   1.597   0.139
## PM2.5Avr     1790.921    3581.381   0.500   0.627
## Age          -2.911     1.721  -1.691   0.119
## BMI          -4.297     3.458  -1.242   0.240
## Sex1        -85.431     63.494  -1.346   0.206
##
## (Dispersion parameter for gaussian family taken to be 8230.99)
##
##      Null deviance: 136071  on 15  degrees of freedom
## Residual deviance:  90541  on 11  degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 195.66
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL25 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL25 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
```

```
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.12503  -0.04893  -0.01916   0.04773   0.26423
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.284914   0.175767   1.621   0.133
## PM2.5Avr     3.262365   4.147534   0.787   0.448
## Age          -0.003055   0.001993  -1.532   0.154
## BMI          -0.005994   0.004005  -1.497   0.163
## Sex1         -0.107006   0.073531  -1.455   0.174
##
## (Dispersion parameter for gaussian family taken to be 0.01103903)
##
##      Null deviance: 0.18851  on 15  degrees of freedom
## Residual deviance: 0.12143  on 11  degrees of freedom
## (26 observations deleted due to missingness)
## AIC: -20.69
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFb1 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFb1 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -52167   -8511    1373   13274   33324
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  65586.6    42598.8   1.540   0.158
## PM2.5Avr     60108.1   103345.4   0.058   0.955
## Age          -220.0      503.4  -0.437   0.672
## BMI           402.4     1030.9   0.390   0.705
## Sex1          5924.6    18493.2   0.320   0.756
##
## (Dispersion parameter for gaussian family taken to be 646809497)
##
##      Null deviance: 6111631239  on 13  degrees of freedom
## Residual deviance: 5821285474  on  9  degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 329.57
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(IL17A ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17A ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## 8.882e-16  8.882e-16  8.882e-16  8.882e-16  8.882e-16
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  5.180e+00  1.855e-15  2.792e+15  <2e-16 ***
## PM2.5Avr      0.000e+00  4.501e-14  0.000e+00      1
## Age           0.000e+00  2.193e-17  0.000e+00      1
## BMI           0.000e+00  4.490e-17  0.000e+00      1
## Sex1          0.000e+00  8.055e-16  0.000e+00      1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.227117e-30)
##
##      Null deviance: 0.0000e+00  on 13  degrees of freedom
## Residual deviance: 1.1044e-29  on  9  degrees of freedom
## (28 observations deleted due to missingness)
## AIC: -918.68
##
## Number of Fisher Scoring iterations: 1

m_cit_PM <- glm(IL1b ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL1b ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.66587  -0.42276  -0.18650  -0.04561   3.04175
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.23133    1.85908   1.200   0.261
## PM2.5Avr     19.30076   45.10169   0.428   0.679
## Age          -0.01408    0.02197  -0.641   0.537
## BMI           0.01047    0.04499   0.233   0.821
## Sex1         -0.45608    0.80707  -0.565   0.586
##

```

```
## (Dispersion parameter for gaussian family taken to be 1.23191)
##
## Null deviance: 12.337 on 13 degrees of freedom
## Residual deviance: 11.087 on 9 degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 48.465
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL6 ~ PM2.5Avr + Age + BMI + Sex,
               family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL6 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -57.01 -28.45 -10.89 18.76 139.04
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) -11.701 94.848 -0.123 0.905
## PM2.5Avr 2193.116 2301.028 0.953 0.365
## Age -0.978 1.121 -0.872 0.406
## BMI 1.215 2.295 0.530 0.609
## Sex1 -23.857 41.176 -0.579 0.577
##
## (Dispersion parameter for gaussian family taken to be 3206.545)
##
## Null deviance: 37387 on 13 degrees of freedom
## Residual deviance: 28859 on 9 degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 158.57
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4RNA ~ PM2.5Avr + Age + BMI + Sex,
               family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4RNA ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
## 3 9 14 16 56 62 63 69
## -0.4826 -0.5029 1.8509 0.4830 -0.3539 -0.2872 0.3539 -0.7409
## 70 82
## -0.1206 -0.1996
```

```
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.27101    2.10625   0.603   0.573
## PM2.5Avr     39.21330   50.86307   0.771   0.476
## Age         -0.02303    0.02837  -0.812   0.454
## BMI          -0.02594    0.05592  -0.464   0.662
## Sex1         -0.84872    1.07567  -0.789   0.466
##
## (Dispersion parameter for gaussian family taken to be 1.016216)
##
## Null deviance: 6.8896  on 9  degrees of freedom
## Residual deviance: 5.0811  on 5  degrees of freedom
## (32 observations deleted due to missingness)
## AIC: 33.608
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17RNA ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17RNA ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##           3           9          14          16          56          62          63
## -0.001980   0.003320  -0.004854   0.004551   0.013508  -0.005896  -0.013508
##          69          70          82
## -0.004089   0.006073   0.002875
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0246387  0.0213237   1.155   0.300
## PM2.5Avr     -0.0543559  0.5149397  -0.106   0.920
## Age         -0.0004068  0.0002872  -1.417   0.216
## BMI          -0.0001444  0.0005661  -0.255   0.809
## Sex1         0.0183142  0.0108902   1.682   0.153
##
## (Dispersion parameter for gaussian family taken to be 0.0001041583)
##
## Null deviance: 0.00133772  on 9  degrees of freedom
## Residual deviance: 0.00052079  on 5  degrees of freedom
## (32 observations deleted due to missingness)
## AIC: -58.249
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFRNA ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)
```

```
##
## Call:
## glm(formula = TGFRNA ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      3      9     14     16     56     62     63
## 69
## -0.02846 -0.60746 -0.27512 -0.21746  0.13549 -0.06636 -0.13549
## 0.38092
##      70      82
## 0.62656 0.18737
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.14238    0.98298   0.145   0.890
## PM2.5Avr     22.49161   23.73769   0.948   0.387
## Age          0.01256    0.01324   0.949   0.386
## BMI         -0.03289    0.02610  -1.260   0.263
## Sex1        -0.67772    0.50201  -1.350   0.235
##
## (Dispersion parameter for gaussian family taken to be 0.221339)
##
## Null deviance: 1.7240  on 9  degrees of freedom
## Residual deviance: 1.1067  on 5  degrees of freedom
## (32 observations deleted due to missingness)
## AIC: 18.367
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(DPP4 ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = DPP4 ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -348.96  -253.46   -14.13   179.90   720.88
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  807.2792   582.1296   1.387   0.1989
## PLogAvr     -3535.2977  93566.4811  -0.038   0.9707
## Age         -0.9461     6.8237  -0.139   0.8928
## BMI          4.6363    14.0146   0.331   0.7484
## Sex1        483.3434   250.3617   1.931   0.0856 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for gaussian family taken to be 121834.8)
##
## Null deviance: 1613202 on 13 degrees of freedom
## Residual deviance: 1096513 on 9 degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 209.49
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TSLP ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TSLP ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -14.279 -8.651 -3.742 0.103 64.968
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.8363 35.7785 0.303 0.768
## PLogAvr 2328.8919 5650.4449 0.412 0.688
## Age -0.3723 0.4051 -0.919 0.378
## BMI 0.2785 0.8167 0.341 0.740
## Sex1 -8.7389 14.9413 -0.585 0.570
##
## (Dispersion parameter for gaussian family taken to be 463.7958)
##
## Null deviance: 5750.5 on 15 degrees of freedom
## Residual deviance: 5101.8 on 11 degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 149.64
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4 ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4 ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -1.3520 -0.6963 -0.2477 0.2229 4.6077
##
## Coefficients:
```



```
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.38495    2.63632  -0.146    0.887
## PLogAvr      279.25681   416.34973   0.671    0.516
## Age          -0.03713    0.02985  -1.244    0.239
## BMI           0.06539    0.06018   1.087    0.300
## Sex1         -0.32150    1.10094  -0.292    0.776
##
## (Dispersion parameter for gaussian family taken to be 2.51813)
##
##      Null deviance: 36.338  on 15  degrees of freedom
## Residual deviance: 27.699  on 11  degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 66.187
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL5 ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL5 ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -21.989   -9.626   -2.785    1.424   56.602
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.1781    34.6256  -0.207    0.840
## PLogAvr      4149.1502   5468.3623   0.759    0.464
## Age          -0.3576     0.3920  -0.912    0.381
## BMI           0.7411     0.7904   0.938    0.369
## Sex1          9.2310    14.4599   0.638    0.536
##
## (Dispersion parameter for gaussian family taken to be 434.3863)
##
##      Null deviance: 6155.4  on 15  degrees of freedom
## Residual deviance: 4778.2  on 11  degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 148.59
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL13 ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL13 ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
```

```
##      data = bacit)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -106.82   -59.70   -25.99    29.77   197.92
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    22.7185    182.0774   0.125   0.903
## PLogAvr      34175.8732  28755.1826   1.189   0.260
## Age           -0.9337     2.0615  -0.453   0.659
## BMI           -0.2666     4.1562  -0.064   0.950
## Sex1          -66.7695    76.0366  -0.878   0.399
##
## (Dispersion parameter for gaussian family taken to be 12011.4)
##
##      Null deviance: 156868  on 15  degrees of freedom
## Residual deviance: 132125  on 11  degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 201.71
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL33 ~ PLogAvr + Age + BMI + Sex,
               family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL33 ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -85.85   -46.24   -14.53    17.41   234.61
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    246.381    150.952   1.632   0.131
## PLogAvr      11085.345  23839.628   0.465   0.651
## Age           -2.974     1.709  -1.740   0.110
## BMI           -4.222     3.446  -1.225   0.246
## Sex1          -83.804    63.039  -1.329   0.211
##
## (Dispersion parameter for gaussian family taken to be 8255.826)
##
##      Null deviance: 136071  on 15  degrees of freedom
## Residual deviance:  90814  on 11  degrees of freedom
## (26 observations deleted due to missingness)
## AIC: 195.71
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(IL25 ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL25 ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.12221  -0.04536  -0.01708   0.04709   0.26535
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.289977   0.174957   1.657   0.126
## PLogAvr      20.768295  27.630676   0.752   0.468
## Age          -0.003165   0.001981  -1.598   0.138
## BMI          -0.005869   0.003994  -1.470   0.170
## Sex1         -0.104381   0.073063  -1.429   0.181
##
## (Dispersion parameter for gaussian family taken to be 0.01109033)
##
##      Null deviance: 0.18851  on 15  degrees of freedom
## Residual deviance: 0.12199  on 11  degrees of freedom
## (26 observations deleted due to missingness)
## AIC: -20.616
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFb1 ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFb1 ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -51211   -9524    1593   13358   34812
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  72855.9    42293.9   1.723   0.119
## PLogAvr     -1596403.9  6797950.0  -0.235   0.820
## Age          -243.1      495.8   -0.490   0.636
## BMI           461.8      1018.2   0.454   0.661
## Sex1          7360.7     18189.7   0.405   0.695
##
## (Dispersion parameter for gaussian family taken to be 643111912)
##

```

```
## Null deviance: 6111631239 on 13 degrees of freedom
## Residual deviance: 5788007212 on 9 degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 329.49
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17A ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17A ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## 8.882e-16  8.882e-16  8.882e-16  8.882e-16  8.882e-16
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.180e+00  1.847e-15  2.804e+15  <2e-16 ***
## PLogAvr      0.000e+00  2.969e-13  0.000e+00      1
## Age          0.000e+00  2.166e-17  0.000e+00      1
## BMI          0.000e+00  4.448e-17  0.000e+00      1
## Sex1         0.000e+00  7.946e-16  0.000e+00      1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.227117e-30)
##
## Null deviance: 0.0000e+00 on 13 degrees of freedom
## Residual deviance: 1.1044e-29 on 9 degrees of freedom
## (28 observations deleted due to missingness)
## AIC: -918.68
##
## Number of Fisher Scoring iterations: 1

m_cit_PM <- glm(IL1b ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL1b ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.64890  -0.41578  -0.19499  -0.06029   3.05502
##
## Coefficients:
```

```
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.28979    1.85510   1.234   0.248
## PLogAvr     112.85846   298.17348   0.378   0.714
## Age         -0.01489    0.02175  -0.685   0.511
## BMI          0.01174    0.04466   0.263   0.799
## Sex1        -0.42940    0.79784  -0.538   0.603
##
## (Dispersion parameter for gaussian family taken to be 1.237282)
##
## Null deviance: 12.337  on 13  degrees of freedom
## Residual deviance: 11.136  on 9  degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 48.525
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL6 ~ PLogAvr + Age + BMI + Sex,
               family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL6 ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -54.98  -29.84  -13.62   19.47  140.62
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -4.499     95.543  -0.047   0.963
## PLogAvr     12670.432  15356.808   0.825   0.431
## Age          -1.071     1.120  -0.956   0.364
## BMI           1.364     2.300   0.593   0.568
## Sex1        -20.719    41.091  -0.504   0.626
##
## (Dispersion parameter for gaussian family taken to be 3281.954)
##
## Null deviance: 37387  on 13  degrees of freedom
## Residual deviance: 29538  on 9  degrees of freedom
## (28 observations deleted due to missingness)
## AIC: 158.89
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4RNA ~ PLogAvr + Age + BMI + Sex,
               family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4RNA ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
```

```
##      data = bacit)
##
## Deviance Residuals:
##      3      9     14     16     56     62     63     69
## -0.4800 -0.3656  1.9166  0.3911 -0.3752 -0.3097  0.3752 -0.7908
##      70     82
## -0.1286 -0.2329
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.35154    2.14218   0.631   0.556
## PLogAvr      199.63489   315.58295   0.633   0.555
## Age           -0.02563    0.02842  -0.902   0.408
## BMI           -0.01779    0.05357  -0.332   0.753
## Sex1          -0.68438    1.02408  -0.668   0.534
##
## (Dispersion parameter for gaussian family taken to be 1.052762)
##
##      Null deviance: 6.8896  on 9  degrees of freedom
## Residual deviance: 5.2638  on 5  degrees of freedom
## (32 observations deleted due to missingness)
## AIC: 33.961
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17RNA ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17RNA ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      3      9     14     16     56     62     63
## -0.002020  0.002832 -0.005147  0.005171  0.013538 -0.005374 -0.013538
##      69     70     82
## -0.004281  0.005813  0.003006
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0240069  0.0213315   1.125   0.312
## PLogAvr      -0.0080273  3.1425223  -0.003   0.998
## Age          -0.0004007  0.0002830  -1.416   0.216
## BMI          -0.0001753  0.0005335  -0.329   0.756
## Sex1         0.0176929  0.0101976   1.735   0.143
##
## (Dispersion parameter for gaussian family taken to be 0.0001043902)
##
##      Null deviance: 0.00133772  on 9  degrees of freedom
## Residual deviance: 0.00052195  on 5  degrees of freedom
## (32 observations deleted due to missingness)
```

```
## AIC: -58.226
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFRNA ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFRNA ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      3          9         14         16         56         62         63
## -0.03702 -0.61146 -0.29383 -0.13275  0.12345  0.05715 -0.12345
## 0.27966
##      70         82
##  0.54626  0.19198
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.04368    0.89869   0.049   0.963
## PLogAvr      189.34741   132.39380   1.430   0.212
## Age           0.01178    0.01192   0.988   0.369
## BMI          -0.03368    0.02247  -1.499   0.194
## Sex1         -0.69308    0.42962  -1.613   0.168
##
## (Dispersion parameter for gaussian family taken to be 0.1852842)
##
##      Null deviance: 1.72405  on 9  degrees of freedom
## Residual deviance: 0.92642  on 5  degrees of freedom
## (32 observations deleted due to missingness)
## AIC: 16.589
##
## Number of Fisher Scoring iterations: 2

#Аллергический фенотип
bacit <- ba[ba$BAcd %in% c("j45.0"), ]
str(bacit)

## 'data.frame':    40 obs. of  106 variables:
## $ ID              : chr  "3" "52" "81" "86" ...
## $ Kazan           : Factor w/ 2 levels "0","1": 1 1 1 2 1 2 2 2 2 2 ...
## $ Point           : Factor w/ 10 levels "1","4","5","6",...: NA NA NA 10 NA 1
## 1 6 6 9 ...
## $ TSPAavr         : num  NA NA NA 0.15 NA 0.164 0.164 0.145 0.145 0.141 ...
## $ TSPMaxAvr       : num  NA NA NA 0.552 NA ...
## $ TSPMaxMax       : num  NA NA NA 0.72 NA 0.734 0.734 0.769 0.769 0.68 ...
## $ PM10Avr         : num  NA NA NA 0.081 NA 0.094 0.094 0.079 0.079 0.074 ...
## $ PM10MaxAvr      : num  NA NA NA 0.277 NA ...
## $ PM10MaxMax      : num  NA NA NA 0.42 NA 0.44 0.44 0.453 0.453 0.42 ...
```

```

## $ PM2.5Avr      : num  NA NA NA 0.025 NA 0.031 0.031 0.024 0.024 0.032 ...
## $ PM2.5MaxAvr   : num  NA NA NA 0.142 NA ...
## $ PM2.5MaxMax   : num  NA NA NA 0.379 NA 0.253 0.253 0.23 0.23 0.23 ...
## $ TSPDT         : num  NA NA NA 0.058 NA 0.038 0.038 0.014 0.014 0.018 ...
## $ PM10DT        : num  NA NA NA 0.046 NA 0.03 0.03 0.011 0.011 0.017 ...
## $ PM25DT        : num  NA NA NA 0.026 NA 0.028 0.028 0.007 0.007 0.012 ...
## $ Cu            : num  NA NA NA NA NA 0.56 0.56 2.39 2.39 NA ...
## $ C             : num  NA NA NA 77.6 NA ...
## $ LAL1          : num  NA NA NA 0.0064 NA 0.0277 0.0277 0.0271 0.0271 NA
...
## $ LAL2          : num  NA NA NA 0.0382 NA 0.0694 0.0694 0.0543 0.0543 NA
...
## $ TBPLogMax     : num  NA NA NA 0.0188 NA ...
## $ PLogMax       : num  NA NA NA 0.0105 NA ...
## $ TBLogMax      : num  NA NA NA 0.00829 NA ...
## $ TBPLogAvr     : num  NA NA NA 0.00759 NA ...
## $ PLogAvr       : num  NA NA NA 0.00429 NA ...
## $ TBLogAvr      : num  NA NA NA 0.0033 NA ...
## $ BA            : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
## $ BAcD          : Factor w/ 4 levels "j45.0","j45.1",...: 1 1 1 1 1 1 1 1 1 1
1 ...
## $ BMI           : num  27.5 25.9 29.4 23.4 NA ...
## $ BMICd        : Factor w/ 3 levels "1","2","3": 2 2 2 1 NA 1 1 1 2 1 ...
## $ BAsevere     : Factor w/ 3 levels "1","2","3": 2 2 3 1 3 2 3 1 2 1 ...
## $ BAobstr      : Factor w/ 2 levels "0","1": 2 1 2 1 2 2 2 1 1 1 ...
## $ BAcontrol    : Factor w/ 4 levels "1","2","3","4": 2 NA 3 1 3 NA NA 4 1
4 ...
## $ BAdebut      : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 1 1 1 1 ...
## $ IgE          : num  155 NA 51.1 555 NA ...
## $ Eos          : num  276 510.3 270 423 97.5 ...
## $ Atopia       : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
## $ Heredity     : Factor w/ 2 levels "0","1": 2 1 1 1 1 2 1 2 1 1 ...
## $ Age          : num  39.7 38.8 46.6 26.4 41.8 ...
## $ Sex          : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 2 2 2 ...
## $ Family       : Factor w/ 6 levels "1","2","3","4",...: 6 6 6 1 5 1 6 5 1
5 ...
## $ FamType      : Factor w/ 5 levels "0","1","2","3",...: 4 4 2 4 2 4 4 4 4
4 ...
## $ Child        : Factor w/ 6 levels "0","1","2","3",...: 2 2 3 1 3 1 2 1 1
1 ...
## $ Child_8      : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 1 1 1 1 ...
## $ Hobby        : Factor w/ 2 levels "0","1": 1 1 2 2 1 1 1 1 2 2 ...
## $ Educ         : Factor w/ 5 levels "2","3","4","5",...: 4 4 1 4 2 1 4 4 4
4 ...
## $ EducYrs      : num  19 17 11 17 13 11.5 18 16 18 15 ...
## $ ScaleS       : Factor w/ 9 levels "1","3","4","5",...: 7 6 6 7 4 4 6 6 7
8 ...
## $ ScaleR       : Factor w/ 10 levels "1","2","3","4",...: 6 5 7 4 5 3 4 6 6
6 ...
## $ ISL_cd       : Factor w/ 4 levels "1","2","3","4": 4 3 2 3 2 2 3 3 4 4
...
## $ ProfCateg    : Factor w/ 5 levels "1","2","3","4",...: 2 NA NA 3 NA 3 3 1

```



```

NA 3 ...
## $ DustCurrent      : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 1 1 1 ...
## $ pDustAn          : int  NA NA NA NA NA NA 0 NA 0 NA ...
## $ Dust             : Factor w/ 2 levels "0","1": 2 1 2 1 2 1 2 1 1 2 ...
## $ ChemCurrent      : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 1 1 1 1 ...
## $ Chem             : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 1 1 1 1 ...
## $ BioCurrent       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ Bio              : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ ColdCurrent      : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 1 1 1 ...
## $ Cold             : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 1 1 1 ...
## $ HeatCurrent      : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 1 1 1 ...
## $ Heat             : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 1 1 1 ...
## $ PhysLoadCurrent  : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 1 2 ...
## $ PhysLoad         : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 2 2 1 2 ...
## $ DL               : int  74 70 46 NA 68 70 52 72 NA 82 ...
## $ JD               : int  21 24 27 NA 26 17 29 26 NA 21 ...
## $ CS               : int  18 20 10 NA 21 17 14 17 NA 20 ...
## $ SS               : int  13 18 9 NA 19 18 14 20 NA 19 ...
## $ SocSup           : int  31 38 19 NA 40 35 28 37 NA 39 ...
## $ JobDis           : num  0.733 0 0.2 NA 0.533 ...
## $ JI               : int  14 11 10 NA 14 17 11 12 NA 13 ...
## $ JDJI             : int  35 35 37 NA 40 34 40 38 NA 34 ...
## $ PE               : int  10 11 16 NA 10 7 11 9 NA 5 ...
## $ JCQedcd          : Factor w/ 2 levels "0","1": 1 1 2 NA 1 1 1 1 NA 1 ...
## $ ERI_Ef_Ph        : int  9 8 20 NA 9 6 6 15 NA 10 ...
## $ ERI_Ef_M         : int  8 7 16 NA 8 5 5 14 NA 9 ...
## $ Overcom          : int  11 7 15 NA 11 7 13 15 NA 15 ...
## $ SANx             : int  33 38 53 NA 52 69 56 46 22 33 ...
## $ SANx_cd          : Factor w/ 3 levels "1","2","3": 2 2 3 NA 3 3 3 3 1 2 ...
## $ LE               : int  108 109 101 NA 53 120 217 311 148 57 ...
## $ LE_cd            : Factor w/ 3 levels "1","2","3": 1 1 1 NA 1 1 2 3 1 1 ...
## $ FamilyPast       : Factor w/ 2 levels "2","3": 2 2 2 2 2 2 2 2 2 1 ...
## $ SMK              : Factor w/ 3 levels "0","1","2": 1 2 1 1 1 1 3 NA NA NA
...
## $ PasSMK           : int  0 4 0 3 0 0 2 NA NA NA ...
## $ PARigMin         : int  240 120 180 120 0 NA NA NA NA NA ...
## $ PAModMin         : int  120 0 180 60 120 NA NA NA NA NA ...
## $ WalkMin          : int  300 240 120 120 180 NA NA NA NA NA ...
## $ FruVegs          : int  0 0 0 0 0 1 1 NA NA NA ...
## $ Omega3           : Factor w/ 2 levels "0","1": 1 1 2 2 2 2 2 NA NA NA ...
## $ DPP4             : num  NA NA NA 1116 NA ...
## $ TGFb1            : num  NA NA NA 57225 NA ...
## $ TSLP             : num  0.485 NA NA 4.62 NA 4.62 4.62 4.62 4.62 4.62 ...
## $ IL13             : num  414.7 NA NA 69.1 NA ...
## $ IL17A            : num  NA NA NA 5.18 NA 5.18 5.18 5.18 5.18 5.18 ...
## $ IL1b             : num  NA NA NA 2.06 NA ...
## $ IL33             : num  14.6 NA NA 200.2 NA ...
## $ IL4              : num  0.005 NA NA 0.005 NA 0.005 0.08 0.49 3.13 0.005 ...
## $ IL5              : num  8.77 NA NA 2.32 NA ...
## $ IL6              : num  NA NA NA 0.505 NA ...
## $ IL25             : num  0.08 NA NA 0.19 NA 0.025 0.025 0.06 0.025 0.025 ...
## [list output truncated]

```

#Обратная связь с ИЛ33 (тенденция), ИЛ25 - PM25Avr, IL25, TGFb1, ИЛ6 - TB/PLogAvr

```
m_cit_PM <- glm(DPP4 ~ PM2.5Avr + Age + BMI + Sex,
                 family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = DPP4 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -472.45  -175.95   -43.82   226.43   393.60
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1394.107   1069.831    1.303   0.215
## PM2.5Avr     -35165.244  27225.298   -1.292   0.219
## Age           -12.807     9.272   -1.381   0.191
## BMI            47.355    29.372    1.612   0.131
## Sex1          -58.831    200.401   -0.294   0.774
##
## (Dispersion parameter for gaussian family taken to be 92401.52)
##
##      Null deviance: 1964388  on 17  degrees of freedom
## Residual deviance: 1201220  on 13  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 263.03
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TSLP ~ PM2.5Avr + Age + BMI + Sex,
                 family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TSLP ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
##  -3.3962  -0.8953  -0.2030   0.4355   5.9586
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.98147    4.94162    0.806   0.433
## PM2.5Avr       37.41803   107.88273    0.347   0.734
## Age           -0.05008    0.05791   -0.865   0.401
## BMI            0.02515    0.18558    0.136   0.894
## Sex1           0.87530    1.30553    0.670   0.513
##
```

```
## (Dispersion parameter for gaussian family taken to be 4.076596)
##
## Null deviance: 79.917 on 19 degrees of freedom
## Residual deviance: 61.149 on 15 degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 91.109
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4 ~ PM2.5Avr + Age + BMI + Sex,
               family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -0.59990 -0.27995 -0.19137 0.01054 2.38753
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.93326 1.84854 0.505 0.621
## PM2.5Avr -42.38947 40.35635 -1.050 0.310
## Age -0.02338 0.02166 -1.079 0.298
## BMI 0.05653 0.06942 0.814 0.428
## Sex1 -0.03240 0.48837 -0.066 0.948
##
## (Dispersion parameter for gaussian family taken to be 0.57045)
##
## Null deviance: 10.2566 on 19 degrees of freedom
## Residual deviance: 8.5568 on 15 degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 51.777
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL5 ~ PM2.5Avr + Age + BMI + Sex,
               family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL5 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -13.812 -6.273 -0.306 2.593 29.398
##
## Coefficients:
```

```
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  45.2881    28.1633   1.608  0.1287
## PM2.5Avr    116.4524   614.8460   0.189  0.8523
## Age         0.5413     0.3301   1.640  0.1218
## BMI        -2.6515     1.0577  -2.507  0.0242 *
## Sex1        7.7478     7.4405   1.041  0.3142
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 132.4118)
##
## Null deviance: 2872.7  on 19  degrees of freedom
## Residual deviance: 1986.2  on 15  degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 160.72
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL13 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL13 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -83.83  -71.14  -23.98    6.64   443.50
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.675e+02  3.157e+02   0.847  0.410
## PM2.5Avr    -8.588e+03  6.893e+03  -1.246  0.232
## Age        -6.751e-03  3.700e+00  -0.002  0.999
## BMI         3.956e+00  1.186e+01   0.334  0.743
## Sex1       -6.742e+01  8.342e+01  -0.808  0.432
##
## (Dispersion parameter for gaussian family taken to be 16642.95)
##
## Null deviance: 306927  on 19  degrees of freedom
## Residual deviance: 249644  on 15  degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 257.4
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL33 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)
```

```
##
## Call:
## glm(formula = IL33 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -47.05  -22.12  -12.21   14.26  119.66
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   118.993    103.777   1.147   0.2695
## PM2.5Avr    -4405.137   2265.614  -1.944   0.0709 .
## Age           -2.926     1.216  -2.406   0.0295 *
## BMI            6.362     3.897   1.632   0.1234
## Sex1         -57.030    27.417  -2.080   0.0551 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1797.901)
##
##      Null deviance: 40581  on 19  degrees of freedom
## Residual deviance: 26969  on 15  degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 212.89
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL25 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL25 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -0.04877  -0.03921  -0.01476   0.02066   0.19356
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.3213684  0.1550211   2.073   0.0558 .
## PM2.5Avr     -9.2146953  3.3843382  -2.723   0.0157 *
## Age          -0.0008823  0.0018167  -0.486   0.6342
## BMI           0.0027119  0.0058219   0.466   0.6480
## Sex1         -0.0787619  0.0409551  -1.923   0.0737 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.00401182)
##
```

```
## Null deviance: 0.112520 on 19 degrees of freedom
## Residual deviance: 0.060177 on 15 degrees of freedom
## (20 observations deleted due to missingness)
## AIC: -47.366
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFb1 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFb1 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -34365  -12457   -943    15577   23286
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.432e+05  6.752e+04   2.121  0.0537 .
## PM2.5Avr     -2.095e+06  1.718e+06  -1.219  0.2444
## Age          -6.895e+01  5.852e+02  -0.118  0.9080
## BMI          -8.295e+02  1.854e+03  -0.447  0.6619
## Sex1         4.534e+03  1.265e+04   0.358  0.7258
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 368070167)
##
## Null deviance: 5706981472 on 17 degrees of freedom
## Residual deviance: 4784912177 on 13 degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 412.25
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17A ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17A ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -5.329e-15  -3.553e-15  -3.109e-15  -2.664e-15  -8.882e-16
##
## Coefficients:
```

```
##           Estimate Std. Error    t value Pr(>|t|)
## (Intercept)  5.180e+00  1.318e-14  3.931e+14   <2e-16 ***
## PM2.5Avr    -2.864e-13  3.353e-13 -8.540e-01   0.409
## Age         -6.322e-17  1.142e-16 -5.540e-01   0.589
## BMI         1.253e-16  3.618e-16  3.460e-01   0.735
## Sex1        -2.431e-15  2.468e-15 -9.850e-01   0.343
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.401745e-29)
##
## Null deviance: 0.0000e+00  on 17  degrees of freedom
## Residual deviance: 1.8223e-28  on 13  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: -1138.6
##
## Number of Fisher Scoring iterations: 1

m_cit_PM <- glm(IL1b ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL1b ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## 0.000e+00  4.441e-16  4.441e-16  7.772e-16  1.332e-15
##
## Coefficients:
##           Estimate Std. Error    t value Pr(>|t|)
## (Intercept)  2.065e+00  2.601e-15  7.940e+14   <2e-16 ***
## PM2.5Avr     7.160e-14  6.619e-14  1.082e+00   0.299
## Age          1.581e-17  2.254e-17  7.010e-01   0.496
## BMI         -3.132e-17  7.141e-17 -4.390e-01   0.668
## Sex1         6.077e-16  4.872e-16  1.247e+00   0.234
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 5.461345e-31)
##
## Null deviance: 0.0000e+00  on 17  degrees of freedom
## Residual deviance: 7.0997e-30  on 13  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: -1197.1
##
## Number of Fisher Scoring iterations: 1

m_cit_PM <- glm(IL6 ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)
```

```
##
## Call:
## glm(formula = IL6 ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -21.502  -14.208   -5.141    3.897   65.124
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   66.9161    86.5853   0.773   0.453
## PM2.5Avr    -2314.9251  2203.4436  -1.051   0.313
## Age           -0.5797     0.7505  -0.772   0.454
## BMI             1.2073     2.3771   0.508   0.620
## Sex1             1.4182    16.2192   0.087   0.932
##
## (Dispersion parameter for gaussian family taken to be 605.254)
##
##      Null deviance: 9947.4  on 17  degrees of freedom
## Residual deviance: 7868.3  on 13  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 172.53
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4RNA ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4RNA ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -0.84218  -0.42346  -0.08417   0.09091   1.84784
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.07338    3.39124  -0.611   0.553
## PM2.5Avr     13.00137    76.64923   0.170   0.868
## Age          -0.04261     0.03078  -1.384   0.194
## BMI           0.13891     0.10989   1.264   0.232
## Sex1         -0.06295     0.57666  -0.109   0.915
##
## (Dispersion parameter for gaussian family taken to be 0.635876)
##
##      Null deviance: 9.3048  on 15  degrees of freedom
## Residual deviance: 6.9946  on 11  degrees of freedom
## (24 observations deleted due to missingness)
## AIC: 44.167
```



```
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17RNA ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17RNA ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3616  -0.4048  -0.0892   0.0596   3.5860
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.06169    5.46061   0.561   0.586
## PM2.5Avr      8.76395   123.42134   0.071   0.945
## Age          -0.02639    0.04956  -0.533   0.605
## BMI          -0.07284    0.17694  -0.412   0.688
## Sex1         -0.89017    0.92854  -0.959   0.358
##
## (Dispersion parameter for gaussian family taken to be 1.648684)
##
##      Null deviance: 24.069  on 15  degrees of freedom
## Residual deviance: 18.136  on 11  degrees of freedom
## (24 observations deleted due to missingness)
## AIC: 59.411
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFRNA ~ PM2.5Avr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFRNA ~ PM2.5Avr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -7.5696  -3.4545  -0.3801   1.4052  12.7078
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -4.1142    25.5962  -0.161   0.8752
## PM2.5Avr     561.6285   578.5288   0.971   0.3525
## Age           0.5329    0.2323   2.294   0.0425 *
## BMI          -1.2078    0.8294  -1.456   0.1733
## Sex1          6.9138    4.3525   1.588   0.1405
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 36.22487)
##
##      Null deviance: 672.62  on 15  degrees of freedom
## Residual deviance: 398.47  on 11  degrees of freedom
## (24 observations deleted due to missingness)
## AIC: 108.85
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(DPP4 ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = DPP4 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -514.25  -160.88   -55.17   238.12   376.08
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.405e+03  1.448e+03   0.970   0.3496
## TBLogAvr     -3.289e+05  3.602e+05  -0.913   0.3779
## Age          -1.497e+01  9.638e+00  -1.553   0.1444
## BMI           5.327e+01  2.959e+01   1.800   0.0951 .
## Sex1         -1.043e+02  2.291e+02  -0.455   0.6564
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 97977.04)
##
##      Null deviance: 1964388  on 17  degrees of freedom
## Residual deviance: 1273702  on 13  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 264.09
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TSLP ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TSLP ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
```

```
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -3.5998  -0.8149   0.0545   0.4137   5.7669
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.766e-01  5.331e+00  -0.071   0.945
## TBLogAvr     1.517e+03  1.165e+03   1.302   0.213
## Age          -2.040e-02  5.872e-02  -0.347   0.733
## BMI          -1.616e-03  1.779e-01  -0.009   0.993
## Sex1         1.412e+00  1.304e+00   1.083   0.296
##
## (Dispersion parameter for gaussian family taken to be 3.692216)
##
##      Null deviance: 79.917  on 19  degrees of freedom
## Residual deviance: 55.383  on 15  degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 89.128
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4 ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.66168  -0.35746  -0.22763   0.04404   2.30000
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.72134    2.05068   0.839   0.414
## TBLogAvr     -604.01254  448.23675  -1.348   0.198
## Age           -0.02988    0.02259  -1.323   0.206
## BMI            0.06590    0.06843   0.963   0.351
## Sex1          -0.16595    0.50155  -0.331   0.745
##
## (Dispersion parameter for gaussian family taken to be 0.5462782)
##
##      Null deviance: 10.2566  on 19  degrees of freedom
## Residual deviance:  8.1942  on 15  degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 50.911
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(IL5 ~ TBLogAvr + Age + BMI + Sex,
                 family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL5 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -14.0952   -5.6874   -0.6566    2.6870   29.0511
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   41.9556    31.8798   1.316   0.2079
## TBLogAvr     1972.9343   6968.2727   0.283   0.7809
## Age           0.5668     0.3511   1.614   0.1273
## BMI          -2.6831     1.0638  -2.522   0.0235 *
## Sex1           8.2482     7.7970   1.058   0.3069
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 132.0229)
##
##      Null deviance: 2872.7  on 19  degrees of freedom
## Residual deviance: 1980.3  on 15  degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 160.66
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL13 ~ TBLogAvr + Age + BMI + Sex,
                 family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL13 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -81.11  -56.80  -26.59   12.82  442.98
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.770e+02  3.533e+02   1.067   0.303
## TBLogAvr     -1.089e+05  7.723e+04  -1.410   0.179
## Age          -9.974e-01  3.892e+00  -0.256   0.801
## BMI           5.600e+00  1.179e+01   0.475   0.642
## Sex1         -8.874e+01  8.641e+01  -1.027   0.321
##

```

```
## (Dispersion parameter for gaussian family taken to be 16215.6)
##
## Null deviance: 306927 on 19 degrees of freedom
## Residual deviance: 243234 on 15 degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 256.88
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL33 ~ TBLLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL33 ~ TBLLogAvr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -51.124 -22.950 -7.429 12.713 136.254
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 92.929 125.448 0.741 0.4703
## TBLLogAvr -33771.749 27420.302 -1.232 0.2370
## Age -2.897 1.382 -2.096 0.0534 .
## BMI 6.790 4.186 1.622 0.1256
## Sex1 -58.565 30.681 -1.909 0.0756 .
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2044.294)
##
## Null deviance: 40581 on 19 degrees of freedom
## Residual deviance: 30664 on 15 degrees of freedom
## (20 observations deleted due to missingness)
## AIC: 215.46
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL25 ~ TBLLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL25 ~ TBLLogAvr + Age + BMI + Sex, family = "gaussian",
## data = bacit)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -0.05833 -0.04302 -0.01185 0.01943 0.18921
```

```
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.367646   0.181170   2.029  0.0606 .
## TBLogAvr    -97.717742  39.599986  -2.468  0.0261 *
## Age         -0.001480   0.001995  -0.742  0.4698
## BMI          0.004116   0.006046   0.681  0.5064
## Sex1        -0.093498   0.044310  -2.110  0.0521 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.004263723)
##
## Null deviance: 0.112520 on 19 degrees of freedom
## Residual deviance: 0.063956 on 15 degrees of freedom
## (20 observations deleted due to missingness)
## AIC: -46.148
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFb1 ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = TGFb1 ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -27721   -9868   -1877   11489   24916
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.502e+05  7.700e+04   3.249  0.00634 **
## TBLogAvr    -4.783e+07  1.916e+07  -2.496  0.02678 *
## Age         -3.127e+02  5.127e+02  -0.610  0.55240
## BMI         -7.191e+02  1.574e+03  -0.457  0.65536
## Sex1        -6.670e+03  1.219e+04  -0.547  0.59352
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 277257284)
##
## Null deviance: 5706981472 on 17 degrees of freedom
## Residual deviance: 3604344688 on 13 degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 407.15
##
## Number of Fisher Scoring iterations: 2
```

```

m_cit_PM <- glm(IL17A ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17A ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -5.329e-15 -4.441e-15 -3.109e-15 -2.664e-15 -1.776e-15
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  5.180e+00  1.943e-14  2.666e+14  <2e-16 ***
## TBLogAvr     -1.588e-12  4.836e-12 -3.280e-01   0.748
## Age          -7.638e-17  1.294e-16 -5.900e-01   0.565
## BMI           1.828e-16  3.973e-16  4.600e-01   0.653
## Sex1         -2.473e-15  3.076e-15 -8.040e-01   0.436
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.765835e-29)
##
##      Null deviance: 0.0000e+00  on 17  degrees of freedom
## Residual deviance: 2.2956e-28  on 13  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: -1134.5
##
## Number of Fisher Scoring iterations: 1

m_cit_PM <- glm(IL1b ~ TBLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL1b ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## 0.000e+00 0.000e+00 4.441e-16 4.441e-16 8.882e-16
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  2.065e+00  2.206e-15  9.361e+14  <2e-16 ***
## TBLogAvr     3.971e-13  5.490e-13  7.230e-01   0.482
## Age          1.910e-17  1.469e-17  1.300e+00   0.216
## BMI          -4.569e-17  4.510e-17 -1.013e+00   0.330
## Sex1         6.183e-16  3.492e-16  1.771e+00   0.100
## ---

```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2.27556e-31)
##
##    Null deviance: 0.0000e+00  on 17  degrees of freedom
## Residual deviance: 2.9582e-30  on 13  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: -1212.8
##
## Number of Fisher Scoring iterations: 1

m_cit_PM <- glm(IL6 ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL6 ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -24.594  -10.918   -6.449    7.168   53.996
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.010e+02  7.363e+01   1.372   0.1934
## PLogAvr      -2.238e+04  1.240e+04  -1.805   0.0943 .
## Age          -8.486e-01  7.074e-01  -1.200   0.2517
## BMI           1.495e+00  2.161e+00   0.692   0.5011
## Sex1         -8.731e+00  1.661e+01  -0.526   0.6080
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 525.0789)
##
##    Null deviance: 9947.4  on 17  degrees of freedom
## Residual deviance: 6826.0  on 13  degrees of freedom
## (22 observations deleted due to missingness)
## AIC: 169.97
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL4RNA ~ PLogAvr + Age + BMI + Sex,
                family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL4RNA ~ PLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
```



```
##      Min      1Q      Median      3Q      Max
## -0.85956 -0.49764 -0.06917  0.10925  1.61988
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.39460    2.98471  -0.132   0.897
## PLogAvr      -218.26204  452.70748  -0.482   0.639
## Age          -0.04269    0.02994  -1.426   0.182
## BMI           0.12574    0.10359   1.214   0.250
## Sex1         -0.20681    0.62874  -0.329   0.748
##
## (Dispersion parameter for gaussian family taken to be 0.6243459)
##
##      Null deviance: 9.3048  on 15  degrees of freedom
## Residual deviance: 6.8678  on 11  degrees of freedom
## (24 observations deleted due to missingness)
## AIC: 43.874
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(IL17RNA ~ TBLogAvr + Age + BMI + Sex,
               family = 'gaussian', data = bacit)
summary(m_cit_PM)

##
## Call:
## glm(formula = IL17RNA ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3941  -0.3116  -0.1048   0.1428   3.5419
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.64348    6.80021   0.830   0.424
## TBLogAvr      -571.17407  1580.94988  -0.361   0.725
## Age           -0.02646    0.04828  -0.548   0.595
## BMI            -0.08848    0.16876  -0.524   0.610
## Sex1          -1.06442    1.02422  -1.039   0.321
##
## (Dispersion parameter for gaussian family taken to be 1.630097)
##
##      Null deviance: 24.069  on 15  degrees of freedom
## Residual deviance: 17.931  on 11  degrees of freedom
## (24 observations deleted due to missingness)
## AIC: 59.229
##
## Number of Fisher Scoring iterations: 2

m_cit_PM <- glm(TGFRNA ~ TBLogAvr + Age + BMI + Sex,
               family = 'gaussian', data = bacit)
summary(m_cit_PM)
```

```
##
## Call:
## glm(formula = TGFRNA ~ TBLogAvr + Age + BMI + Sex, family = "gaussian",
##      data = bacit)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -6.632  -3.558  -1.064   3.479  11.810
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -25.6631    30.2484  -0.848   0.4143
## TBLogAvr     10923.8723  7032.3136   1.553   0.1486
## Age           0.5943     0.2147   2.768   0.0183 *
## BMI          -1.2591     0.7507  -1.677   0.1216
## Sex1          9.5427     4.5559   2.095   0.0602 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 32.25326)
##
##      Null deviance: 672.62  on 15  degrees of freedom
## Residual deviance: 354.79  on 11  degrees of freedom
## (24 observations deleted due to missingness)
## AIC: 106.99
##
## Number of Fisher Scoring iterations: 2

"Интегральная оценка цитокинового профиля методом главных компонент"

## [1] "Интегральная оценка цитокинового профиля методом главных компонент"

d <- data.frame(ba$DPP4, ba$TSLP, ba$IL4, ba$IL5, ba$IL6, ba$IL13,
               ba$IL17A, ba$IL25, ba$IL33, ba$IL1b, ba$TGFB1, ba$BAcd, ba$PLogAvr)
d <- na.omit(d)
str(d)

## 'data.frame':    71 obs. of  13 variables:
## $ ba.DPP4      : num  926 1232 1116 1069 1199 ...
## $ ba.TSLP      : num  4.62 4.62 4.62 4.62 4.62 ...
## $ ba.IL4       : num  1.31 0.005 0.005 0.005 0.005 0.08 6.21 1.66 0.005 0.03
## ...
## $ ba.IL5       : num  22.51 2.32 2.32 2.32 7.44 ...
## $ ba.IL6       : num  46.05 0.505 0.505 1.4 0.505 ...
## $ ba.IL13      : num  102.3 28.07 69.07 40.73 3.18 ...
## $ ba.IL17A     : num  5.18 5.18 5.18 5.18 5.18 5.18 5.18 5.18 5.18 5.18 ...
## $ ba.IL25      : num  0.025 0.025 0.19 0.025 0.025 0.025 0.025 0.025 0.025
0.025 ...
## $ ba.IL33      : num  5.65 1.79 200.24 1.79 6.48 ...
## $ ba.IL1b      : num  2.06 2.06 2.06 2.06 2.06 ...
## $ ba.TGFB1     : num  76425 71332 57225 99988 56434 ...
## $ ba.BAcd      : Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4 ...
## $ ba.PLogAvr   : num  0.00457 0.00447 0.00429 0.00457 0.00457 ...
```

```
## - attr(*, "na.action")= 'omit' Named int [1:70] 1 2 4 6 7 10 11 17 18 20 ...
## .. attr(*, "names")= chr [1:70] "1" "2" "4" "6" ...

d1 <- d[1:11]
str(d1)

## 'data.frame': 71 obs. of 11 variables:
## $ ba.DPP4 : num 926 1232 1116 1069 1199 ...
## $ ba.TSLP : num 4.62 4.62 4.62 4.62 4.62 ...
## $ ba.IL4 : num 1.31 0.005 0.005 0.005 0.005 0.08 6.21 1.66 0.005 0.03 ...
## $ ba.IL5 : num 22.51 2.32 2.32 2.32 7.44 ...
## $ ba.IL6 : num 46.05 0.505 0.505 1.4 0.505 ...
## $ ba.IL13 : num 102.3 28.07 69.07 40.73 3.18 ...
## $ ba.IL17A: num 5.18 5.18 5.18 5.18 5.18 5.18 5.18 5.18 5.18 5.18 ...
## $ ba.IL25 : num 0.025 0.025 0.19 0.025 0.025 0.025 0.025 0.025 0.025 0.025 ...
## $ ba.IL33 : num 5.65 1.79 200.24 1.79 6.48 ...
## $ ba.IL1b : num 2.06 2.06 2.06 2.06 2.06 ...
## $ ba.TGFb1: num 76425 71332 57225 99988 56434 ...

pca <- prcomp(d1, center = TRUE, scale = TRUE)
pca

## Standard deviations (1, ..., p=11):
## [1] 2.47279911 1.49037926 1.03544156 0.97638373 0.56015290 0.43099870
## [7] 0.24236090 0.21066213 0.15129690 0.10821467 0.03633395
##
## Rotation (n x k) = (11 x 11):
##
```

	PC1	PC2	PC3	PC4	PC5
ba.DPP4	0.01545410	0.04933365	0.80751007	-0.540116653	-0.22373224
ba.TSLP	0.35421818	-0.30670573	0.02999345	0.018292880	-0.05553489
ba.IL4	0.31419658	0.39902655	0.02216487	0.002851536	0.25905876
ba.IL5	0.27186613	0.34245627	-0.21485770	0.074330513	-0.85795476
ba.IL6	0.33932030	0.33572809	0.05954571	0.021463443	0.28122356
ba.IL13	0.29018242	0.43944505	0.03423700	-0.040594390	0.18119523
ba.IL17A	0.35198960	-0.28505139	0.06127450	0.002356539	0.10421071
ba.IL25	0.34297653	-0.34320827	0.02953653	0.040304722	-0.06789312
ba.IL33	0.33610364	-0.34131212	0.02419662	0.063876009	-0.07058843
ba.IL1b	0.38073125	-0.04185925	-0.03372320	-0.016458416	0.09006219
ba.TGFb1	0.05351956	-0.06230273	-0.53787098	-0.833256613	0.05639392

```
##
```

	PC6	PC7	PC8	PC9	PC10
ba.DPP4	0.02423541	0.004753663	0.03733893	-0.037440644	0.007922452
ba.TSLP	-0.16869462	0.122806643	0.32958620	0.672779753	0.013678816
ba.IL4	-0.12687795	0.321191233	0.38832235	-0.235053547	-0.590858939
ba.IL5	0.07359629	0.101888965	-0.06741870	-0.023888689	-0.003154949
ba.IL6	-0.03240542	0.409982756	-0.10526634	-0.046263299	0.711776511
ba.IL13	-0.25845316	-0.615567460	-0.37239460	0.300102653	-0.111723610
ba.IL17A	0.39897792	0.282610118	-0.65448162	0.002795971	-0.319720705
ba.IL25	-0.27461193	0.028587389	0.04512827	0.064125715	-0.006422136
ba.IL33	-0.41456099	-0.241182160	-0.03130656	-0.616789403	0.104472115
ba.IL1b	0.68676953	-0.428578879	0.39148551	-0.113691215	0.133823500
ba.TGFb1	-0.05902453	0.042511145	-0.01602255	-0.021635221	0.020789275

```
##
## PC11
```

```
## ba.DPP4 -0.002003172
## ba.TSLP 0.413710769
## ba.IL4 0.005190113
## ba.IL5 0.010658327
## ba.IL6 -0.008500437
## ba.IL13 -0.009358521
## ba.IL17A 0.102965285
## ba.IL25 -0.821614244
## ba.IL33 0.372207861
## ba.IL1b -0.065913729
## ba.TGFb1 0.001970699
```

```
pca$rotation
```

```
##          PC1          PC2          PC3          PC4          PC5
## ba.DPP4 0.01545410 0.04933365 0.80751007 -0.540116653 -0.22373224
## ba.TSLP 0.35421818 -0.30670573 0.02999345 0.018292880 -0.05553489
## ba.IL4 0.31419658 0.39902655 0.02216487 0.002851536 0.25905876
## ba.IL5 0.27186613 0.34245627 -0.21485770 0.074330513 -0.85795476
## ba.IL6 0.33932030 0.33572809 0.05954571 0.021463443 0.28122356
## ba.IL13 0.29018242 0.43944505 0.03423700 -0.040594390 0.18119523
## ba.IL17A 0.35198960 -0.28505139 0.06127450 0.002356539 0.10421071
## ba.IL25 0.34297653 -0.34320827 0.02953653 0.040304722 -0.06789312
## ba.IL33 0.33610364 -0.34131212 0.02419662 0.063876009 -0.07058843
## ba.IL1b 0.38073125 -0.04185925 -0.03372320 -0.016458416 0.09006219
## ba.TGFb1 0.05351956 -0.06230273 -0.53787098 -0.833256613 0.05639392
##          PC6          PC7          PC8          PC9          PC10
## ba.DPP4 0.02423541 0.004753663 0.03733893 -0.037440644 0.007922452
## ba.TSLP -0.16869462 0.122806643 0.32958620 0.672779753 0.013678816
## ba.IL4 -0.12687795 0.321191233 0.38832235 -0.235053547 -0.590858939
## ba.IL5 0.07359629 0.101888965 -0.06741870 -0.023888689 -0.003154949
## ba.IL6 -0.03240542 0.409982756 -0.10526634 -0.046263299 0.711776511
## ba.IL13 -0.25845316 -0.615567460 -0.37239460 0.300102653 -0.111723610
## ba.IL17A 0.39897792 0.282610118 -0.65448162 0.002795971 -0.319720705
## ba.IL25 -0.27461193 0.028587389 0.04512827 0.064125715 -0.006422136
## ba.IL33 -0.41456099 -0.241182160 -0.03130656 -0.616789403 0.104472115
## ba.IL1b 0.68676953 -0.428578879 0.39148551 -0.113691215 0.133823500
## ba.TGFb1 -0.05902453 0.042511145 -0.01602255 -0.021635221 0.020789275
##          PC11
## ba.DPP4 -0.002003172
## ba.TSLP 0.413710769
## ba.IL4 0.005190113
## ba.IL5 0.010658327
## ba.IL6 -0.008500437
## ba.IL13 -0.009358521
## ba.IL17A 0.102965285
## ba.IL25 -0.821614244
## ba.IL33 0.372207861
## ba.IL1b -0.065913729
## ba.TGFb1 0.001970699
```

```
summary(pca)
```

```
## Importance of components:
##               PC1      PC2      PC3      PC4      PC5      PC6      PC7
## Standard deviation  2.4728 1.4904 1.03544 0.97638 0.56015 0.43100 0.24236
## Proportion of Variance 0.5559 0.2019 0.09747 0.08667 0.02852 0.01689 0.00534
## Cumulative Proportion 0.5559 0.7578 0.85528 0.94195 0.97047 0.98736 0.99270
##               PC8      PC9      PC10     PC11
## Standard deviation  0.21066 0.15130 0.10821 0.03633
## Proportion of Variance 0.00403 0.00208 0.00106 0.00012
## Cumulative Proportion 0.99673 0.99882 0.99988 1.00000
```

```
plot(pca, type = "lines")
m <- predict(pca, newdata = d1)
m <- as.data.frame(m)
```

```
with2 <- data.frame(m$PC1,d$ba.BAcid)
#colnames(with2)[ncol(with2)] <- "PC2"
with2
```

```
##           m.PC1 d.ba.BAcid
## 1    0.69133808      j45.1
## 2   -0.92520572      j45.8
## 3   -0.33764819      j45.0
## 4   -0.78493069      j45.1
## 5   -0.99232380      j45.0
## 6    0.13832124      j45.0
## 7    7.69684385      j45.1
## 8    1.13866291      j45.8
## 9   -0.85402036      j45.1
## 10  -0.82235200         K
## 11  -0.93101744         K
## 12  -0.56538512         K
## 13  -1.13622376         K
## 14  -0.99644257         K
## 15  -0.55241037         K
## 16  -0.92725871         K
## 17  -0.39787416         K
## 18  -0.40211477         K
## 19  -0.81581423         K
## 20  -1.05984376         K
## 21  -0.31397619         K
## 22  -0.02688057      j45.0
## 23  -0.59341583         K
## 24   2.02679915      j45.0
## 25  -0.87222950      j45.0
## 26  -0.87986749      j45.0
## 27  -0.97873886         K
## 28  -0.80248870         K
## 29  -0.94119822         K
## 30  -0.84465483         K
## 31   0.55367126         K
## 32  -0.86120949      j45.1
## 33   0.34473819      j45.1
## 34  -0.08352245      j45.1
```

```
## 35  0.30427451      j45.0
## 36 -1.05673543      j45.1
## 37 -0.15927715      j45.1
## 38 -0.98636315      j45.0
## 39 -0.95819096      j45.0
## 40 -0.13780011      j45.0
## 41  0.52671080      j45.1
## 42 -1.10916511      j45.0
## 43 -0.95371685      j45.1
## 44 -1.07694250      j45.1
## 45 -0.95237563      j45.8
## 46 -0.27769648      j45.8
## 47 -0.66791037      j45.0
## 48 -0.79960446      j45.0
## 49  0.34566799      j45.1
## 50 -0.89246482      K
## 51  1.52922003      K
## 52  0.12042612      K
## 53 -0.35686495      K
## 54  0.03785996      K
## 55 -0.92670568      j45.0
## 56 -0.83744289      K
## 57 -0.84752080      j45.0
## 58 17.43620998      K
## 59  4.04074915      K
## 60 -0.82215855      K
## 61 -0.94411864      K
## 62 -0.75181167      K
## 63 -0.77264506      K
## 64 -0.82271052      K
## 65 -0.63771022      K
## 66  0.72101530      K
## 67  1.33211139      K
## 68 -1.09616885      j45.0
## 69 -0.90875050      j45.0
## 70  1.40532484      j45.1
## 71 -0.93804562      j45.8
```

```
str(with2)
```

```
## 'data.frame':    71 obs. of  2 variables:
## $ m.PC1      : num  0.691 -0.925 -0.338 -0.785 -0.992 ...
## $ d.ba.BAcd: Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4 ...
```

```
with2_ba <- with2[with2$d.ba.BAcd %in% c("K", "j45.0"), ]
str(with2_ba)
```

```
## 'data.frame':    52 obs. of  2 variables:
## $ m.PC1      : num  -0.338 -0.992 0.138 -0.822 -0.931 ...
## $ d.ba.BAcd: Factor w/ 4 levels "j45.0","j45.1",...: 1 1 1 4 4 4 4 4 4 4 ...
```

```

m_cit_PC <- glm(m.PC1 ~ d.ba.BAcD,
                family = 'gaussian', data = with2_ba)
summary(m_cit_PC)

##
## Call:
## glm(formula = m.PC1 ~ d.ba.BAcD, family = "gaussian", data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3083  -0.9943  -0.5495  -0.1136   17.2641
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.5043     0.6244  -0.808    0.423
## d.ba.BAcDK    0.6764     0.7722   0.876    0.385
##
## (Dispersion parameter for gaussian family taken to be 7.018278)
##
##      Null deviance: 356.30  on 51  degrees of freedom
## Residual deviance: 350.91  on 50  degrees of freedom
## AIC: 252.85
##
## Number of Fisher Scoring iterations: 2

with2 <- data.frame(m$PC1,d$ba.BAcD)
#colnames(with2)[ncol(with2)] <- "PC2"
with2

##      m.PC1 d.ba.BAcD
## 1  0.69133808      j45.1
## 2 -0.92520572      j45.8
## 3 -0.33764819      j45.0
## 4 -0.78493069      j45.1
## 5 -0.99232380      j45.0
## 6  0.13832124      j45.0
## 7  7.69684385      j45.1
## 8  1.13866291      j45.8
## 9 -0.85402036      j45.1
## 10 -0.82235200         K
## 11 -0.93101744         K
## 12 -0.56538512         K
## 13 -1.13622376         K
## 14 -0.99644257         K
## 15 -0.55241037         K
## 16 -0.92725871         K
## 17 -0.39787416         K
## 18 -0.40211477         K
## 19 -0.81581423         K
## 20 -1.05984376         K
## 21 -0.31397619         K
## 22 -0.02688057      j45.0
## 23 -0.59341583         K

```

##	24	2.02679915	j45.0
##	25	-0.87222950	j45.0
##	26	-0.87986749	j45.0
##	27	-0.97873886	K
##	28	-0.80248870	K
##	29	-0.94119822	K
##	30	-0.84465483	K
##	31	0.55367126	K
##	32	-0.86120949	j45.1
##	33	0.34473819	j45.1
##	34	-0.08352245	j45.1
##	35	0.30427451	j45.0
##	36	-1.05673543	j45.1
##	37	-0.15927715	j45.1
##	38	-0.98636315	j45.0
##	39	-0.95819096	j45.0
##	40	-0.13780011	j45.0
##	41	0.52671080	j45.1
##	42	-1.10916511	j45.0
##	43	-0.95371685	j45.1
##	44	-1.07694250	j45.1
##	45	-0.95237563	j45.8
##	46	-0.27769648	j45.8
##	47	-0.66791037	j45.0
##	48	-0.79960446	j45.0
##	49	0.34566799	j45.1
##	50	-0.89246482	K
##	51	1.52922003	K
##	52	0.12042612	K
##	53	-0.35686495	K
##	54	0.03785996	K
##	55	-0.92670568	j45.0
##	56	-0.83744289	K
##	57	-0.84752080	j45.0
##	58	17.43620998	K
##	59	4.04074915	K
##	60	-0.82215855	K
##	61	-0.94411864	K
##	62	-0.75181167	K
##	63	-0.77264506	K
##	64	-0.82271052	K
##	65	-0.63771022	K
##	66	0.72101530	K
##	67	1.33211139	K
##	68	-1.09616885	j45.0
##	69	-0.90875050	j45.0
##	70	1.40532484	j45.1
##	71	-0.93804562	j45.8

str(with2)


```
## 'data.frame': 71 obs. of 2 variables:
## $ m.PC1 : num 0.691 -0.925 -0.338 -0.785 -0.992 ...
## $ d.ba.BAcd: Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4 ...

with2_ba <- with2[with2$d.ba.BAcd %in% c("K", "j45.1"), ]
str(with2_ba)

## 'data.frame': 48 obs. of 2 variables:
## $ m.PC1 : num 0.691 -0.785 7.697 -0.854 -0.822 ...
## $ d.ba.BAcd: Factor w/ 4 levels "j45.0","j45.1",...: 2 2 2 2 4 4 4 4 4 4 ...

m_cit_PC <- glm(m.PC1 ~ d.ba.BAcd,
               family = 'gaussian', data = with2_ba)
summary(m_cit_PC)

##
## Call:
## glm(formula = m.PC1 ~ d.ba.BAcd, family = "gaussian", data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.4470  -1.1057  -0.8669  -0.0451  17.2641
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.3700     0.7944   0.466   0.644
## d.ba.BAcdK    -0.1979     0.9439  -0.210   0.835
##
## (Dispersion parameter for gaussian family taken to be 8.834925)
##
##      Null deviance: 406.79  on 47  degrees of freedom
## Residual deviance: 406.41  on 46  degrees of freedom
## AIC: 244.75
##
## Number of Fisher Scoring iterations: 2

with2 <- data.frame(m$PC1,d$ba.BAcd)
#colnames(with2)[ncol(with2)] <- "PC2"
with2

##           m.PC1 d.ba.BAcd
## 1  0.69133808      j45.1
## 2 -0.92520572      j45.8
## 3 -0.33764819      j45.0
## 4 -0.78493069      j45.1
## 5 -0.99232380      j45.0
## 6  0.13832124      j45.0
## 7  7.69684385      j45.1
## 8  1.13866291      j45.8
## 9 -0.85402036      j45.1
## 10 -0.82235200         K
## 11 -0.93101744         K
## 12 -0.56538512         K
## 13 -1.13622376         K
```

## 14	-0.99644257	K
## 15	-0.55241037	K
## 16	-0.92725871	K
## 17	-0.39787416	K
## 18	-0.40211477	K
## 19	-0.81581423	K
## 20	-1.05984376	K
## 21	-0.31397619	K
## 22	-0.02688057	j45.0
## 23	-0.59341583	K
## 24	2.02679915	j45.0
## 25	-0.87222950	j45.0
## 26	-0.87986749	j45.0
## 27	-0.97873886	K
## 28	-0.80248870	K
## 29	-0.94119822	K
## 30	-0.84465483	K
## 31	0.55367126	K
## 32	-0.86120949	j45.1
## 33	0.34473819	j45.1
## 34	-0.08352245	j45.1
## 35	0.30427451	j45.0
## 36	-1.05673543	j45.1
## 37	-0.15927715	j45.1
## 38	-0.98636315	j45.0
## 39	-0.95819096	j45.0
## 40	-0.13780011	j45.0
## 41	0.52671080	j45.1
## 42	-1.10916511	j45.0
## 43	-0.95371685	j45.1
## 44	-1.07694250	j45.1
## 45	-0.95237563	j45.8
## 46	-0.27769648	j45.8
## 47	-0.66791037	j45.0
## 48	-0.79960446	j45.0
## 49	0.34566799	j45.1
## 50	-0.89246482	K
## 51	1.52922003	K
## 52	0.12042612	K
## 53	-0.35686495	K
## 54	0.03785996	K
## 55	-0.92670568	j45.0
## 56	-0.83744289	K
## 57	-0.84752080	j45.0
## 58	17.43620998	K
## 59	4.04074915	K
## 60	-0.82215855	K
## 61	-0.94411864	K
## 62	-0.75181167	K
## 63	-0.77264506	K
## 64	-0.82271052	K
## 65	-0.63771022	K

```

## 66  0.72101530      K
## 67  1.33211139      K
## 68 -1.09616885    j45.0
## 69 -0.90875050    j45.0
## 70  1.40532484    j45.1
## 71 -0.93804562    j45.8

str(with2)

## 'data.frame':    71 obs. of  2 variables:
## $ m.PC1      : num  0.691 -0.925 -0.338 -0.785 -0.992 ...
## $ d.ba.BACd: Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4 ...

with2_ba <- with2[with2$d.ba.BACd %in% c("j45.1", "j45.0"), ]
str(with2_ba)

## 'data.frame':    32 obs. of  2 variables:
## $ m.PC1      : num  0.691 -0.338 -0.785 -0.992 0.138 ...
## $ d.ba.BACd: Factor w/ 4 levels "j45.0","j45.1",...: 2 1 2 1 1 2 2 1 1 1 ...

m_cit_PC <- glm(m.PC1 ~ d.ba.BACd,
                family = 'gaussian', data = with2_ba)
summary(m_cit_PC)

##
## Call:
## glm(formula = m.PC1 ~ d.ba.BACd, family = "gaussian", data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.4470  -0.5449  -0.3717   0.2053   7.3268
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.5043     0.3744  -1.347   0.188
## d.ba.BACdj45.1  0.8743     0.5661   1.545   0.133
##
## (Dispersion parameter for gaussian family taken to be 2.5235)
##
##      Null deviance: 81.725  on 31  degrees of freedom
## Residual deviance: 75.705  on 30  degrees of freedom
## AIC: 124.37
##
## Number of Fisher Scoring iterations: 2

with2 <- data.frame(m$PC2,d$ba.BACd)
#colnames(with2)[ncol(with2)] <- "PC2"
with2

##           m.PC2 d.ba.BACd
## 1  1.39973360    j45.1
## 2 -0.56532370    j45.8
## 3 -0.59903192    j45.0
## 4 -0.55653809    j45.1

```

## 5	-0.61613208	j45.0
## 6	0.72916798	j45.0
## 7	7.58895223	j45.1
## 8	2.12943418	j45.8
## 9	-0.64545052	j45.1
## 10	-0.33840721	K
## 11	-0.49079590	K
## 12	-0.22532982	K
## 13	-0.67251038	K
## 14	-0.55315953	K
## 15	-0.05254058	K
## 16	-0.63368117	K
## 17	-0.22054917	K
## 18	-0.17267819	K
## 19	-0.19360015	K
## 20	-0.52899759	K
## 21	0.04549853	K
## 22	0.23204864	j45.0
## 23	-0.16019965	K
## 24	3.11168539	j45.0
## 25	-0.54139485	j45.0
## 26	-0.49592037	j45.0
## 27	-0.52082626	K
## 28	-0.38371984	K
## 29	-0.66625217	K
## 30	-0.40209012	K
## 31	1.23914823	K
## 32	-0.51555365	j45.1
## 33	-1.02114045	j45.1
## 34	0.35098127	j45.1
## 35	1.21323785	j45.0
## 36	-0.76252803	j45.1
## 37	0.35434285	j45.1
## 38	-0.60724696	j45.0
## 39	-0.38262763	j45.0
## 40	0.38545602	j45.0
## 41	1.35839650	j45.1
## 42	-0.84087588	j45.0
## 43	-0.64221058	j45.1
## 44	-0.46279789	j45.1
## 45	-0.86129397	j45.8
## 46	0.08934325	j45.8
## 47	-0.21751463	j45.0
## 48	-0.78193964	j45.0
## 49	1.04569790	j45.1
## 50	-0.50509513	K
## 51	2.56129099	K
## 52	-0.36914071	K
## 53	0.55394945	K
## 54	0.63670068	K
## 55	-0.64907755	j45.0
## 56	-0.53844914	K

```

## 57 -0.33426627      j45.0
## 58 -6.18569691      K
## 59  1.87775073      K
## 60 -0.34598978      K
## 61 -0.62125212      K
## 62 -0.47189645      K
## 63 -0.47214119      K
## 64 -0.52304657      K
## 65 -0.27417793      K
## 66 -0.57277135      K
## 67  2.14091681      K
## 68 -0.74835772      j45.0
## 69 -0.66711193      j45.0
## 70  2.45366956      j45.1
## 71 -0.88807334      j45.8

str(with2)

## 'data.frame':    71 obs. of  2 variables:
## $ m.PC2      : num  1.4 -0.565 -0.599 -0.557 -0.616 ...
## $ d.ba.BAcd: Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4 ...

with2_ba <- with2[with2$d.ba.BAcd %in% c("K", "j45.0"), ]
str(with2_ba)

## 'data.frame':    52 obs. of  2 variables:
## $ m.PC2      : num  -0.599 -0.616 0.729 -0.338 -0.491 ...
## $ d.ba.BAcd: Factor w/ 4 levels "j45.0","j45.1",...: 1 1 1 4 4 4 4 4 4 4 ...

m_cit_PC <- glm(m.PC2 ~ d.ba.BAcd,
                family = 'gaussian', data = with2_ba)
summary(m_cit_PC)

##
## Call:
## glm(formula = m.PC2 ~ d.ba.BAcd, family = "gaussian", data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -5.9492  -0.3958  -0.2346   0.1032   3.2122
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.1006     0.2887  -0.348   0.729
## d.ba.BAcdK   -0.1359     0.3570  -0.381   0.705
##
## (Dispersion parameter for gaussian family taken to be 1.500163)
##
##      Null deviance: 75.226  on 51  degrees of freedom
## Residual deviance: 75.008  on 50  degrees of freedom
## AIC: 172.62
##
## Number of Fisher Scoring iterations: 2

```

```
with2 <- data.frame(m$PC2,d$ba.BAcid)
#colnames(with2)[ncol(with2)] <- "PC2"
with2
```

```
##          m.PC2 d.ba.BAcid
## 1    1.39973360    j45.1
## 2   -0.56532370    j45.8
## 3   -0.59903192    j45.0
## 4   -0.55653809    j45.1
## 5   -0.61613208    j45.0
## 6    0.72916798    j45.0
## 7    7.58895223    j45.1
## 8    2.12943418    j45.8
## 9   -0.64545052    j45.1
## 10  -0.33840721      K
## 11  -0.49079590      K
## 12  -0.22532982      K
## 13  -0.67251038      K
## 14  -0.55315953      K
## 15  -0.05254058      K
## 16  -0.63368117      K
## 17  -0.22054917      K
## 18  -0.17267819      K
## 19  -0.19360015      K
## 20  -0.52899759      K
## 21   0.04549853      K
## 22   0.23204864    j45.0
## 23  -0.16019965      K
## 24   3.11168539    j45.0
## 25  -0.54139485    j45.0
## 26  -0.49592037    j45.0
## 27  -0.52082626      K
## 28  -0.38371984      K
## 29  -0.66625217      K
## 30  -0.40209012      K
## 31   1.23914823      K
## 32  -0.51555365    j45.1
## 33  -1.02114045    j45.1
## 34   0.35098127    j45.1
## 35   1.21323785    j45.0
## 36  -0.76252803    j45.1
## 37   0.35434285    j45.1
## 38  -0.60724696    j45.0
## 39  -0.38262763    j45.0
## 40   0.38545602    j45.0
## 41   1.35839650    j45.1
## 42  -0.84087588    j45.0
## 43  -0.64221058    j45.1
## 44  -0.46279789    j45.1
## 45  -0.86129397    j45.8
## 46   0.08934325    j45.8
## 47  -0.21751463    j45.0
```

```

## 48 -0.78193964      j45.0
## 49  1.04569790      j45.1
## 50 -0.50509513      K
## 51  2.56129099      K
## 52 -0.36914071      K
## 53  0.55394945      K
## 54  0.63670068      K
## 55 -0.64907755      j45.0
## 56 -0.53844914      K
## 57 -0.33426627      j45.0
## 58 -6.18569691      K
## 59  1.87775073      K
## 60 -0.34598978      K
## 61 -0.62125212      K
## 62 -0.47189645      K
## 63 -0.47214119      K
## 64 -0.52304657      K
## 65 -0.27417793      K
## 66 -0.57277135      K
## 67  2.14091681      K
## 68 -0.74835772      j45.0
## 69 -0.66711193      j45.0
## 70  2.45366956      j45.1
## 71 -0.88807334      j45.8

str(with2)

## 'data.frame':    71 obs. of  2 variables:
## $ m.PC2      : num  1.4 -0.565 -0.599 -0.557 -0.616 ...
## $ d.ba.BAcD: Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4 ...

with2_ba <- with2[with2$d.ba.BAcD %in% c("K", "j45.1"), ]
str(with2_ba)

## 'data.frame':    48 obs. of  2 variables:
## $ m.PC2      : num  1.4 -0.557 7.589 -0.645 -0.338 ...
## $ d.ba.BAcD: Factor w/ 4 levels "j45.0","j45.1",...: 2 2 2 2 4 4 4 4 4 4 ...

m_cit_PC <- glm(m.PC2 ~ d.ba.BAcD,
                family = 'gaussian', data = with2_ba)
summary(m_cit_PC)

##
## Call:
## glm(formula = m.PC2 ~ d.ba.BAcD, family = "gaussian", data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -5.9492  -0.3658  -0.2005   0.2084   6.8786
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.7104     0.4380   1.622   0.1116
## d.ba.BAcDK    -0.9469     0.5204  -1.819   0.0754 .

```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2.685694)
##
##      Null deviance: 132.43  on 47  degrees of freedom
## Residual deviance: 123.54  on 46  degrees of freedom
## AIC: 187.6
##
## Number of Fisher Scoring iterations: 2
```

```
with2 <- data.frame(m$PC2,d$ba.BAcD)
#colnames(with2)[ncol(with2)] <- "PC2"
with2
```

##	m.PC2	d.ba.BAcD
## 1	1.39973360	j45.1
## 2	-0.56532370	j45.8
## 3	-0.59903192	j45.0
## 4	-0.55653809	j45.1
## 5	-0.61613208	j45.0
## 6	0.72916798	j45.0
## 7	7.58895223	j45.1
## 8	2.12943418	j45.8
## 9	-0.64545052	j45.1
## 10	-0.33840721	K
## 11	-0.49079590	K
## 12	-0.22532982	K
## 13	-0.67251038	K
## 14	-0.55315953	K
## 15	-0.05254058	K
## 16	-0.63368117	K
## 17	-0.22054917	K
## 18	-0.17267819	K
## 19	-0.19360015	K
## 20	-0.52899759	K
## 21	0.04549853	K
## 22	0.23204864	j45.0
## 23	-0.16019965	K
## 24	3.11168539	j45.0
## 25	-0.54139485	j45.0
## 26	-0.49592037	j45.0
## 27	-0.52082626	K
## 28	-0.38371984	K
## 29	-0.66625217	K
## 30	-0.40209012	K
## 31	1.23914823	K
## 32	-0.51555365	j45.1
## 33	-1.02114045	j45.1
## 34	0.35098127	j45.1
## 35	1.21323785	j45.0
## 36	-0.76252803	j45.1
## 37	0.35434285	j45.1


```

## 38 -0.60724696      j45.0
## 39 -0.38262763      j45.0
## 40  0.38545602      j45.0
## 41  1.35839650      j45.1
## 42 -0.84087588      j45.0
## 43 -0.64221058      j45.1
## 44 -0.46279789      j45.1
## 45 -0.86129397      j45.8
## 46  0.08934325      j45.8
## 47 -0.21751463      j45.0
## 48 -0.78193964      j45.0
## 49  1.04569790      j45.1
## 50 -0.50509513      K
## 51  2.56129099      K
## 52 -0.36914071      K
## 53  0.55394945      K
## 54  0.63670068      K
## 55 -0.64907755      j45.0
## 56 -0.53844914      K
## 57 -0.33426627      j45.0
## 58 -6.18569691      K
## 59  1.87775073      K
## 60 -0.34598978      K
## 61 -0.62125212      K
## 62 -0.47189645      K
## 63 -0.47214119      K
## 64 -0.52304657      K
## 65 -0.27417793      K
## 66 -0.57277135      K
## 67  2.14091681      K
## 68 -0.74835772      j45.0
## 69 -0.66711193      j45.0
## 70  2.45366956      j45.1
## 71 -0.88807334      j45.8

str(with2)

## 'data.frame':    71 obs. of  2 variables:
## $ m.PC2      : num  1.4 -0.565 -0.599 -0.557 -0.616 ...
## $ d.ba.BAcd: Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4 ...

with2_ba <- with2[with2$d.ba.BAcd %in% c("j45.1", "j45.0"), ]
str(with2_ba)

## 'data.frame':    32 obs. of  2 variables:
## $ m.PC2      : num  1.4 -0.599 -0.557 -0.616 0.729 ...
## $ d.ba.BAcd: Factor w/ 4 levels "j45.0","j45.1",...: 2 1 2 1 1 2 2 1 1 1 ...

m_cit_PC <- glm(m.PC2 ~ d.ba.BAcd,
                family = 'gaussian', data = with2_ba)
summary(m_cit_PC)

##
## Call:

```

```
## glm(formula = m.PC2 ~ d.ba.BACd, family = "gaussian", data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.7315   -0.6961   -0.4181    0.3730    6.8786
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.1006     0.3881  -0.259    0.797
## d.ba.BACdj45.1  0.8109     0.5868   1.382    0.177
##
## (Dispersion parameter for gaussian family taken to be 2.711343)
##
##      Null deviance: 86.519  on 31  degrees of freedom
## Residual deviance: 81.340  on 30  degrees of freedom
## AIC: 126.67
##
## Number of Fisher Scoring iterations: 2

with2 <- data.frame(m$PC2,d$ba.BACd)
#colnames(with2)[ncol(with2)] <- "PC2"
with2

##           m.PC2 d.ba.BACd
## 1  1.39973360    j45.1
## 2 -0.56532370    j45.8
## 3 -0.59903192    j45.0
## 4 -0.55653809    j45.1
## 5 -0.61613208    j45.0
## 6  0.72916798    j45.0
## 7  7.58895223    j45.1
## 8  2.12943418    j45.8
## 9 -0.64545052    j45.1
## 10 -0.33840721      K
## 11 -0.49079590      K
## 12 -0.22532982      K
## 13 -0.67251038      K
## 14 -0.55315953      K
## 15 -0.05254058      K
## 16 -0.63368117      K
## 17 -0.22054917      K
## 18 -0.17267819      K
## 19 -0.19360015      K
## 20 -0.52899759      K
## 21  0.04549853      K
## 22  0.23204864    j45.0
## 23 -0.16019965      K
## 24  3.11168539    j45.0
## 25 -0.54139485    j45.0
## 26 -0.49592037    j45.0
## 27 -0.52082626      K
## 28 -0.38371984      K
## 29 -0.66625217      K
```

```

## 30 -0.40209012      K
## 31  1.23914823      K
## 32 -0.51555365     j45.1
## 33 -1.02114045     j45.1
## 34  0.35098127     j45.1
## 35  1.21323785     j45.0
## 36 -0.76252803     j45.1
## 37  0.35434285     j45.1
## 38 -0.60724696     j45.0
## 39 -0.38262763     j45.0
## 40  0.38545602     j45.0
## 41  1.35839650     j45.1
## 42 -0.84087588     j45.0
## 43 -0.64221058     j45.1
## 44 -0.46279789     j45.1
## 45 -0.86129397     j45.8
## 46  0.08934325     j45.8
## 47 -0.21751463     j45.0
## 48 -0.78193964     j45.0
## 49  1.04569790     j45.1
## 50 -0.50509513      K
## 51  2.56129099      K
## 52 -0.36914071      K
## 53  0.55394945      K
## 54  0.63670068      K
## 55 -0.64907755     j45.0
## 56 -0.53844914      K
## 57 -0.33426627     j45.0
## 58 -6.18569691      K
## 59  1.87775073      K
## 60 -0.34598978      K
## 61 -0.62125212      K
## 62 -0.47189645      K
## 63 -0.47214119      K
## 64 -0.52304657      K
## 65 -0.27417793      K
## 66 -0.57277135      K
## 67  2.14091681      K
## 68 -0.74835772     j45.0
## 69 -0.66711193     j45.0
## 70  2.45366956     j45.1
## 71 -0.88807334     j45.8

str(with2)

## 'data.frame': 71 obs. of 2 variables:
## $ m.PC2 : num 1.4 -0.565 -0.599 -0.557 -0.616 ...
## $ d.ba.BAcd: Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4 ...

with2_ba <- with2[with2$d.ba.BAcd %in% c("j45.1", "j45.0", "K"), ]
str(with2_ba)

```

```
## 'data.frame': 66 obs. of 2 variables:
## $ m.PC2 : num 1.4 -0.599 -0.557 -0.616 0.729 ...
## $ d.ba.BAcD: Factor w/ 4 levels "j45.0","j45.1",...: 2 1 2 1 1 2 2 4 4 4 ...
```

```
m_cit_PC <- glm(m.PC2 ~ d.ba.BAcD,
               family = 'gaussian', data = with2_ba)
summary(m_cit_PC)
```

```
##
## Call:
## glm(formula = m.PC2 ~ d.ba.BAcD, family = "gaussian", data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -5.9492  -0.4396  -0.2450   0.2575   6.8786
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.1006     0.3513  -0.286   0.776
## d.ba.BAcDj45.1  0.8109     0.5311   1.527   0.132
## d.ba.BAcDK     -0.1359     0.4344  -0.313   0.755
##
## (Dispersion parameter for gaussian family taken to be 2.221352)
##
##      Null deviance: 149.09  on 65  degrees of freedom
## Residual deviance: 139.95  on 63  degrees of freedom
## AIC: 244.91
##
## Number of Fisher Scoring iterations: 2
```

##PM и цитокиновый статус при различных фенотипах

```
with2 <- data.frame(m$PC1,d$ba.BAcD, d$ba.PLogAvr)
#colnames(with2)[ncol(with2)] <- "PC2"
with2
```

```
##      m.PC1 d.ba.BAcD d.ba.PLogAvr
## 1  0.69133808      j45.1 0.004568806
## 2 -0.92520572      j45.8 0.004472117
## 3 -0.33764819      j45.0 0.004286372
## 4 -0.78493069      j45.1 0.004568806
## 5 -0.99232380      j45.0 0.004568806
## 6  0.13832124      j45.0 0.004568806
## 7  7.69684385      j45.1 0.004568806
## 8  1.13866291      j45.8 0.004472117
## 9 -0.85402036      j45.1 0.001509666
## 10 -0.82235200       K 0.004472117
## 11 -0.93101744       K 0.001812206
## 12 -0.56538512       K 0.004472117
## 13 -1.13622376       K 0.004472117
## 14 -0.99644257       K 0.003089037
## 15 -0.55241037       K 0.003089037
## 16 -0.92725871       K 0.001544791
```

## 17	-0.39787416	K	0.003089037
## 18	-0.40211477	K	0.004286372
## 19	-0.81581423	K	0.001812206
## 20	-1.05984376	K	0.004286372
## 21	-0.31397619	K	0.001812206
## 22	-0.02688057	j45.0	0.003089037
## 23	-0.59341583	K	0.001812206
## 24	2.02679915	j45.0	0.003089037
## 25	-0.87222950	j45.0	0.004286372
## 26	-0.87986749	j45.0	0.004568806
## 27	-0.97873886	K	0.004286372
## 28	-0.80248870	K	0.004568806
## 29	-0.94119822	K	0.004568806
## 30	-0.84465483	K	0.004151693
## 31	0.55367126	K	0.002272403
## 32	-0.86120949	j45.1	0.004568806
## 33	0.34473819	j45.1	0.004568806
## 34	-0.08352245	j45.1	0.004568806
## 35	0.30427451	j45.0	0.004568806
## 36	-1.05673543	j45.1	0.001544791
## 37	-0.15927715	j45.1	0.004568806
## 38	-0.98636315	j45.0	0.004568806
## 39	-0.95819096	j45.0	0.004568806
## 40	-0.13780011	j45.0	0.004568806
## 41	0.52671080	j45.1	0.004568806
## 42	-1.10916511	j45.0	0.004568806
## 43	-0.95371685	j45.1	0.004286372
## 44	-1.07694250	j45.1	0.004286372
## 45	-0.95237563	j45.8	0.004286372
## 46	-0.27769648	j45.8	0.004286372
## 47	-0.66791037	j45.0	0.004286372
## 48	-0.79960446	j45.0	0.004151693
## 49	0.34566799	j45.1	0.004568806
## 50	-0.89246482	K	0.004568806
## 51	1.52922003	K	0.004472117
## 52	0.12042612	K	0.004568806
## 53	-0.35686495	K	0.004472117
## 54	0.03785996	K	0.004568806
## 55	-0.92670568	j45.0	0.004568806
## 56	-0.83744289	K	0.004286372
## 57	-0.84752080	j45.0	0.004151693
## 58	17.43620998	K	0.004286372
## 59	4.04074915	K	0.004286372
## 60	-0.82215855	K	0.003089037
## 61	-0.94411864	K	0.004151693
## 62	-0.75181167	K	0.004472117
## 63	-0.77264506	K	0.002272403
## 64	-0.82271052	K	0.001544791
## 65	-0.63771022	K	0.002272403
## 66	0.72101530	K	0.004286372
## 67	1.33211139	K	0.001544791
## 68	-1.09616885	j45.0	0.004568806

```
## 69 -0.90875050      j45.0  0.003089037
## 70  1.40532484      j45.1  0.004568806
## 71 -0.93804562      j45.8  0.004568806

str(with2)

## 'data.frame':    71 obs. of  3 variables:
## $ m.PC1          : num  0.691 -0.925 -0.338 -0.785 -0.992 ...
## $ d.ba.BAcd      : Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4
## $ d.ba.PLogAvr   : num  0.00457 0.00447 0.00429 0.00457 0.00457 ...

with2_ba <- with2[with2$d.ba.BAcd %in% c("K", "j45.1", "j45.0"), ]
str(with2_ba)

## 'data.frame':    66 obs. of  3 variables:
## $ m.PC1          : num  0.691 -0.338 -0.785 -0.992 0.138 ...
## $ d.ba.BAcd      : Factor w/ 4 levels "j45.0","j45.1",...: 2 1 2 1 1 2 2 4 4 4
## $ d.ba.PLogAvr   : num  0.00457 0.00429 0.00457 0.00457 0.00457 ...

m_cit_PC <- glm(m.PC1 ~ d.ba.PLogAvr + d.ba.BAcd,
                family = 'gaussian', data = with2_ba)
summary(m_cit_PC)

##
## Call:
## glm(formula = m.PC1 ~ d.ba.PLogAvr + d.ba.BAcd, family = "gaussian",
##      data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.6338   -0.9868   -0.4908   -0.0075   17.0009
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.9218     1.4754  -1.303    0.198
## d.ba.PLogAvr  335.2114    318.1561   1.054    0.296
## d.ba.BAcdj45.1  0.9195     0.9165   1.003    0.320
## d.ba.BAcdK     0.9203     0.7838   1.174    0.245
##
## (Dispersion parameter for gaussian family taken to be 6.59978)
##
##      Null deviance: 423.96  on 65  degrees of freedom
## Residual deviance: 409.19  on 62  degrees of freedom
## AIC: 317.72
##
## Number of Fisher Scoring iterations: 2

##Основная модель!!! - сниженная реакция по T2 туну!!!
with2 <- data.frame(m$PC2,d$ba.BAcd, d$ba.PLogAvr)
#colnames(with2)[ncol(with2)] <- "PC2"
with2
```

##	m.PC2	d.ba.BAcd	d.ba.PLogAvr
## 1	1.39973360	j45.1	0.004568806
## 2	-0.56532370	j45.8	0.004472117
## 3	-0.59903192	j45.0	0.004286372
## 4	-0.55653809	j45.1	0.004568806
## 5	-0.61613208	j45.0	0.004568806
## 6	0.72916798	j45.0	0.004568806
## 7	7.58895223	j45.1	0.004568806
## 8	2.12943418	j45.8	0.004472117
## 9	-0.64545052	j45.1	0.001509666
## 10	-0.33840721	K	0.004472117
## 11	-0.49079590	K	0.001812206
## 12	-0.22532982	K	0.004472117
## 13	-0.67251038	K	0.004472117
## 14	-0.55315953	K	0.003089037
## 15	-0.05254058	K	0.003089037
## 16	-0.63368117	K	0.001544791
## 17	-0.22054917	K	0.003089037
## 18	-0.17267819	K	0.004286372
## 19	-0.19360015	K	0.001812206
## 20	-0.52899759	K	0.004286372
## 21	0.04549853	K	0.001812206
## 22	0.23204864	j45.0	0.003089037
## 23	-0.16019965	K	0.001812206
## 24	3.11168539	j45.0	0.003089037
## 25	-0.54139485	j45.0	0.004286372
## 26	-0.49592037	j45.0	0.004568806
## 27	-0.52082626	K	0.004286372
## 28	-0.38371984	K	0.004568806
## 29	-0.66625217	K	0.004568806
## 30	-0.40209012	K	0.004151693
## 31	1.23914823	K	0.002272403
## 32	-0.51555365	j45.1	0.004568806
## 33	-1.02114045	j45.1	0.004568806
## 34	0.35098127	j45.1	0.004568806
## 35	1.21323785	j45.0	0.004568806
## 36	-0.76252803	j45.1	0.001544791
## 37	0.35434285	j45.1	0.004568806
## 38	-0.60724696	j45.0	0.004568806
## 39	-0.38262763	j45.0	0.004568806
## 40	0.38545602	j45.0	0.004568806
## 41	1.35839650	j45.1	0.004568806
## 42	-0.84087588	j45.0	0.004568806
## 43	-0.64221058	j45.1	0.004286372
## 44	-0.46279789	j45.1	0.004286372
## 45	-0.86129397	j45.8	0.004286372
## 46	0.08934325	j45.8	0.004286372
## 47	-0.21751463	j45.0	0.004286372
## 48	-0.78193964	j45.0	0.004151693
## 49	1.04569790	j45.1	0.004568806
## 50	-0.50509513	K	0.004568806
## 51	2.56129099	K	0.004472117

```

## 52 -0.36914071      K  0.004568806
## 53  0.55394945      K  0.004472117
## 54  0.63670068      K  0.004568806
## 55 -0.64907755    j45.0 0.004568806
## 56 -0.53844914      K  0.004286372
## 57 -0.33426627    j45.0 0.004151693
## 58 -6.18569691      K  0.004286372
## 59  1.87775073      K  0.004286372
## 60 -0.34598978      K  0.003089037
## 61 -0.62125212      K  0.004151693
## 62 -0.47189645      K  0.004472117
## 63 -0.47214119      K  0.002272403
## 64 -0.52304657      K  0.001544791
## 65 -0.27417793      K  0.002272403
## 66 -0.57277135      K  0.004286372
## 67  2.14091681      K  0.001544791
## 68 -0.74835772    j45.0 0.004568806
## 69 -0.66711193    j45.0 0.003089037
## 70  2.45366956    j45.1 0.004568806
## 71 -0.88807334    j45.8 0.004568806

str(with2)

## 'data.frame':    71 obs. of  3 variables:
## $ m.PC2          : num  1.4 -0.565 -0.599 -0.557 -0.616 ...
## $ d.ba.BAcd       : Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4
## ...
## $ d.ba.PLogAvr: num  0.00457 0.00447 0.00429 0.00457 0.00457 ...

with2_ba <- with2[with2$d.ba.BAcd %in% c("K", "j45.1", "j45.0"), ]
str(with2_ba)

## 'data.frame':    66 obs. of  3 variables:
## $ m.PC2          : num  1.4 -0.599 -0.557 -0.616 0.729 ...
## $ d.ba.BAcd       : Factor w/ 4 levels "j45.0","j45.1",...: 2 1 2 1 1 2 2 4 4 4
## ...
## $ d.ba.PLogAvr: num  0.00457 0.00429 0.00457 0.00457 0.00457 ...

m_cit_PC <- glm(m.PC2 ~ d.ba.PLogAvr * d.ba.BAcd,
                family = 'gaussian', data = with2_ba)
summary(m_cit_PC)

##
## Call:
## glm(formula = m.PC2 ~ d.ba.PLogAvr * d.ba.BAcd, family = "gaussian",
##      data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -5.8333  -0.4124  -0.1717   0.1466   6.5933
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.005      2.792   1.076  0.2860

```



```
## d.ba.PLogAvr          -734.400      654.999   -1.121    0.2667
## d.ba.BAcj45.1         -4.754        3.210   -1.481    0.1438
## d.ba.BAcK             -2.725        2.908   -0.937    0.3525
## d.ba.PLogAvr:d.ba.BAcj45.1 1335.164    754.616    1.769    0.0819 .
## d.ba.PLogAvr:d.ba.BAcK      586.773    691.247    0.849    0.3993
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2.177322)
##
##      Null deviance: 149.09  on 65  degrees of freedom
## Residual deviance: 130.64  on 60  degrees of freedom
## AIC: 246.36
##
## Number of Fisher Scoring iterations: 2

with2 <- data.frame(m$PC3,d$ba.BAcj, d$ba.PLogAvr)
#colnames(with2)[ncol(with2)] <- "PC2"
with2

##           m.PC3 d.ba.BAcj d.ba.PLogAvr
## 1  -0.874092408      j45.1  0.004568806
## 2   0.204317484      j45.8  0.004472117
## 3   0.309131867      j45.0  0.004286372
## 4  -0.900452778      j45.1  0.004568806
## 5   0.385624501      j45.0  0.004568806
## 6  -2.744577322      j45.0  0.004568806
## 7  -1.788962369      j45.1  0.004568806
## 8  -0.071429984      j45.8  0.004472117
## 9  -0.338036374      j45.1  0.001509666
## 10  1.581774808         K  0.004472117
## 11 -0.294161206         K  0.001812206
## 12 -1.423899122         K  0.004472117
## 13  0.363057440         K  0.004472117
## 14  1.351076012         K  0.003089037
## 15 -0.228110205         K  0.003089037
## 16 -0.825676524         K  0.001544791
## 17  0.266189692         K  0.003089037
## 18  1.027343924         K  0.004286372
## 19  2.347146242         K  0.001812206
## 20  1.497823492         K  0.004286372
## 21  0.022595790         K  0.001812206
## 22 -0.379202410      j45.0  0.003089037
## 23 -0.270974971         K  0.001812206
## 24  1.488999509      j45.0  0.003089037
## 25 -0.387874725      j45.0  0.004286372
## 26  0.182479351      j45.0  0.004568806
## 27 -0.333900538         K  0.004286372
## 28  1.408352408         K  0.004568806
## 29 -0.714429608         K  0.004568806
## 30  1.357061664         K  0.004151693
## 31 -0.486735472         K  0.002272403
## 32 -0.512051148      j45.1  0.004568806
```

```
## 33 -1.862900668      j45.1  0.004568806
## 34 -1.549402604      j45.1  0.004568806
## 35  2.012304366      j45.0  0.004568806
## 36 -1.071701041      j45.1  0.001544791
## 37 -0.033164549      j45.1  0.004568806
## 38 -0.298413467      j45.0  0.004568806
## 39  0.944868159      j45.0  0.004568806
## 40 -0.408551447      j45.0  0.004568806
## 41  1.818207686      j45.1  0.004568806
## 42 -1.139303647      j45.0  0.004568806
## 43 -0.864480151      j45.1  0.004286372
## 44 -0.045004091      j45.1  0.004286372
## 45  0.006680968      j45.8  0.004286372
## 46 -0.701717257      j45.8  0.004286372
## 47  1.355644005      j45.0  0.004286372
## 48 -1.385853574      j45.0  0.004151693
## 49 -0.424784311      j45.1  0.004568806
## 50  0.063391140              K  0.004568806
## 51 -0.334521373              K  0.004472117
## 52 -0.549964220              K  0.004568806
## 53  2.148328793              K  0.004472117
## 54  0.166659023              K  0.004568806
## 55 -0.239543169      j45.0  0.004568806
## 56 -0.408550419              K  0.004286372
## 57  0.049262720      j45.0  0.004151693
## 58  0.300472808              K  0.004286372
## 59  0.796005829              K  0.004286372
## 60  0.578917671              K  0.003089037
## 61 -0.004393346              K  0.004151693
## 62 -0.389046977              K  0.004472117
## 63 -0.615207381              K  0.002272403
## 64  0.691196662              K  0.001544791
## 65  0.376438223              K  0.002272403
## 66  1.271019763              K  0.004286372
## 67  0.686380965              K  0.001544791
## 68 -0.499756178      j45.0  0.004568806
## 69 -0.283691095      j45.0  0.003089037
## 70  1.004757263      j45.1  0.004568806
## 71 -2.378992102      j45.8  0.004568806
```

```
str(with2)
```

```
## 'data.frame':    71 obs. of  3 variables:
##  $ m.PC3          : num  -0.874 0.204 0.309 -0.9 0.386 ...
##  $ d.ba.BAcd      : Factor w/ 4 levels "j45.0","j45.1",...: 2 3 1 2 1 1 2 3 2 4
##  ...
##  $ d.ba.PLogAvr: num  0.00457 0.00447 0.00429 0.00457 0.00457 ...
```

```
with2_ba <- with2[with2$d.ba.BAcd %in% c("K", "j45.1", "j45.0"), ]
str(with2_ba)
```

```
## 'data.frame':    66 obs. of  3 variables:
##  $ m.PC3          : num  -0.874 0.309 -0.9 0.386 -2.745 ...
```

```
## $ d.ba.BAcD : Factor w/ 4 levels "j45.0","j45.1",...: 2 1 2 1 1 2 2 4 4 4
...
## $ d.ba.PLogAvr: num 0.00457 0.00429 0.00457 0.00457 0.00457 ...

m_cit_PC <- glm(m.PC3 ~ d.ba.PLogAvr + d.ba.BAcD,
                family = 'gaussian', data = with2_ba)
summary(m_cit_PC)

##
## Call:
## glm(formula = m.PC3 ~ d.ba.PLogAvr + d.ba.BAcD, family = "gaussian",
##      data = with2_ba)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.6973  -0.5725  -0.1917   0.4695   2.3352
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.1873     0.5694  -0.329   0.743
## d.ba.PLogAvr   30.6529    122.7776   0.250   0.804
## d.ba.BAcDj45.1 -0.4698     0.3537  -1.328   0.189
## d.ba.BAcDK      0.4159     0.3025   1.375   0.174
##
## (Dispersion parameter for gaussian family taken to be 0.9828508)
##
##      Null deviance: 68.720  on 65  degrees of freedom
## Residual deviance: 60.937  on 62  degrees of freedom
## AIC: 192.03
##
## Number of Fisher Scoring iterations: 2
```

