CWRU Biostatistics Assignment

Eric Liu

2023-03-01

Stage 1

Download data from NHANES website (Stage 1)

```
# install.packages(foreign)
library(foreign)
# use foreign package to transfer xpt file from CDC website to R
# download diabetes from NHANES 2017 to 2018 Demographics Data
download.file("https://wwwn.cdc.gov/Nchs/Nhanes/2017-2018/DIQ_J.XPT", t1 <- tempfile(), mode = "wb")
diabetes = foreign::read.xport(t1)[,c("SEQN","DIQ010")]
# download diabetes from NHANES 2017 to 2018 Questionnaire Data-Diabetes
download.file("https://wwwn.cdc.gov/Nchs/Nhanes/2017-2018/DEMO_J.XPT", t2 <- tempfile(), mode = "wb")
population = foreign::read.xport(t2)[,c("SEQN","RIAGENDR","RIDRETH3","INDHHIN2")]
```

Create database with diabetes and population

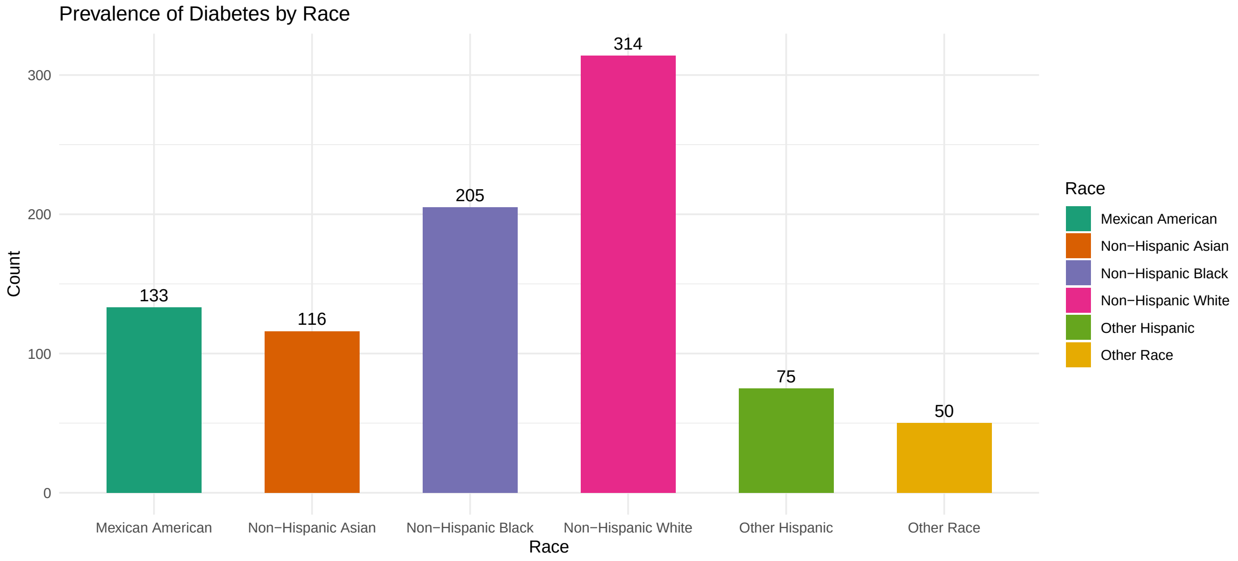
```
# dplyr function can perform left join as well
# library(dplyr)
# left_join(dia, race, by = "SEQN")

# left join diabetes database and population database, since diabetes has less observations
dia_pop = merge(x = diabetes, y = population, by = "SEQN", all.x = T)
# rename database
names(dia_pop) = c("id", "Diabetes", "Gender", "Race", "Annual household income")
```

Barplot of race and number of each race

```
# filter out the data set with observation has diabetes
m1 = which(dia_pop$Diabetes %in% c(1))
diabetes_new = dia_pop[c(m1), ]
# create a data frame with 6 different races
dia_race = as.data.frame(table(diabetes_new$Race))
dia_race[, 1] = c("Mexican American", "Other Hispanic", "Non-Hispanic White",
                  "Non-Hispanic Black", "Non-Hispanic Asian", "Other Race")
names(dia_race) = c("Race", "Count")
# create a bar plot of race and count
```

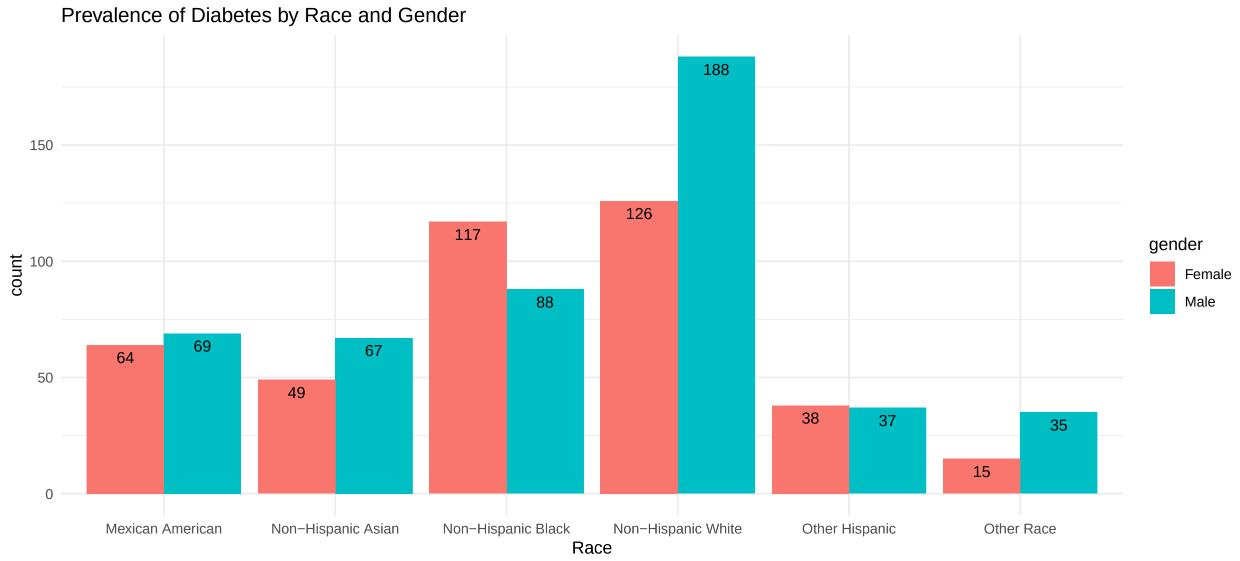
```r
# install.packages(ggplot2)
library(ggplot2)
p1 = ggplot(dia_race, aes(x = Race, y = Count, fill = Race)) + ggtitle("Prevalence of Diabetes by Race")
     geom_bar(stat = "identity", width = 0.6) +
     geom_text(aes(label = Count), vjust = -0.5, size = 4) + theme_minimal()
p1 + scale_fill_brewer(palette = "Dark2")
```



Barplot of race and gender

```r
# a2_1 = which(diabetes_new$Race %in% c(1))
# dia_race_1 = diabetes_new[c(a2_1), ]
# as.data.frame(table(dia_race_1$Gender))

# create a data frame of genders and races
gender = c("Male","Female","Male","Female","Male","Female",
           "Male","Female","Male","Female","Male","Female")
count  = c(69,64,37,38,188,126,88,117,67,49,35,15)
race_1 = c("Mexican American","Mexican American","Other Hispanic","Other Hispanic",
           "Non-Hispanic White","Non-Hispanic White","Non-Hispanic Black","Non-Hispanic Black",
           "Non-Hispanic Asian","Non-Hispanic Asian","Other Race","Other Race")
dia_race_gender = data.frame(gender, race_1, count)
# create a bar plot of race and gender
p2 = ggplot(dia_race_gender, aes(x = race_1, y = count, fill = gender)) +
     ggtitle("Prevalence of Diabetes by Race and Gender") + xlab("Race") +
     geom_bar(stat = "identity", position = position_dodge()) +
     geom_text(aes(label=count), vjust = 1.6,
     position = position_dodge(0.9), size = 3.5) + theme_minimal()
p2
```



Interpretation: For 893 individuals in our study who have diagnosed diabetes, 409 (45.8%) are female, and 484 (54.2%) are male (Table 1). After taking the total population from the demographics data into consideration, we can find out that female (8.7%) has lower prevalence of diabetes compared with male (10.6%). This pattern also is observed among Mexican American, non-Hispanic Asian, non-Hispanic White, and other race, except for non-Hispanic Black, and other Hispanic. For non-Hispanic Black, female percentage (28.6%) is much higher than male percentage (18.2%). In addition, for other Hispanic, female percentage (9.3) is slightly higher than male percentage (7.6%). Overall, other race has the lowest prevalence of diabetes (8.3%) compared with Mexican American (8.7%), other Hispanic (9.6%), non-Hispanic White (10.4%), non-Hispanic Black (10%), and non-Hispanic Asian (10%). This pattern is observed between both female and male. The prevalence of diabetes was not statistically different for other Hispanic, non-Hispanic white, and non-Hispanic black, and non-Hispanic Asian adults.

Stage 2

Download data from NHANES website (Stage 2)

```
# download diabetes from NHANES 2017 to 2018 Questionnaire Data-sleeping disorder
download.file("https://wwwn.cdc.gov/Nchs/Nhanes/2017-2018/SLQ_J.XPT", t3 <- tempfile(), mode = "wb")
sleep_disorder = foreign::read.xport(t3)[,c("SEQN","SLQ050")]
# download diabetes from NHANES 2017 to 2018 Questionnaire Data-cigarettes use
download.file("https://wwwn.cdc.gov/Nchs/Nhanes/2017-2018/SMQ_J.XPT", t4 <- tempfile(), mode = "wb")
smoke = foreign::read.xport(t4)[,c("SEQN","SMQ040")]
# download diabetes from NHANES 2017 to 2018 Questionnaire Data-alcohol use
download.file("https://wwwn.cdc.gov/Nchs/Nhanes/2017-2018/ALQ_J.XPT", t5 <- tempfile(), mode = "wb")
alcohol = foreign::read.xport(t5)[,c("SEQN","ALQ121")]
```

Create database with diabetes and gender, race, annual household income, sleep disorders, cigarettes use, and alcohol use

```
# left join diabetes data frame with population, sleep disorders, cigarettes use,
# and alcohol use by turn
final = merge(x = diabetes, y = population, by = "SEQN", all.x = T)
```

```r
final = merge(x = final, y = sleep_disorder, by = "SEQN", all.x = T)
final = merge(x = final, y = smoke, by = "SEQN", all.x = T)
final = merge(x = final, y = alcohol, by = "SEQN", all.x = T)
# delete the first column since it is Questionnaire id
final = final[,-1]
names(final) = c("Diabetes", "Gender", "Race", "Income", "Sleeping_disorder", "Cigarettes", "Alcohol")
```

Read and preprocess data for final database

```r
a1_1 = which(final$Diabetes %in% c(1,3)) # Diabetes & Prediabetes
final[c(a1_1), 1] = 1
a1_2 = which(final$Diabetes %in% c(7,9)) # Get rid of Don't know and Refused
final = final[-a1_2,]

b1_1 = which(final$Income %in% c(1,2,3,4,13)) # Annual household income under $20,000
final[c(b1_1), 4] = 1
b1_2 = which(final$Income %in% c(5,6,7,8,9,12)) # Annual household income $20,000 to $64,999
final[c(b1_2), 4] = 2
b1_3 = which(final$Income %in% c(10,14)) # Annual household income $65,000 to $99,999
final[c(b1_3), 4] = 3
b1_4 = which(final$Income %in% c(15)) # Annual household income $100,000 and over
final[c(b1_4), 4] = 4
b1_5 = which(final$Income %in% c(77,99)) # Get rid of Don't know and Refused
final = final[-b1_5,]

c1_1 = which(final$Sleeping_disorder %in% c(7,9)) # Get rid of Don't know and Refused
final = final[-c1_1,]

d1_1 = which(final$Cigarettes %in% c(1,2)) # Active smoker
final[c(d1_1), 6] = 1
d1_1 = which(final$Cigarettes %in% c(3)) # Adjust the code
final[c(d1_1), 6] = 2

e1_1 = which(final$Alcohol %in% c(1,2)) # Drink alcohol nearly every day
final[c(e1_1), 7] = 1
e1_2 = which(final$Alcohol %in% c(3,4,5)) # Drink alcohol at least once a week
final[c(e1_2), 7] = 2
e1_3 = which(final$Alcohol %in% c(6,7,8)) # Drink alcohol at least once a month
final[c(e1_3), 7] = 3
e1_4 = which(final$Alcohol %in% c(9,10)) # Drink alcohol at least once a year
final[c(e1_4), 7] = 4
e1_5 = which(final$Alcohol %in% c(0)) # Never drink
final[c(e1_5), 7] = 5
e1_6 = which(final$Alcohol %in% c(77,99)) # Get rid of Don't know and Refused
final = final[-e1_6,]

f1_1 = which(final$Race %in% c(6)) # Adjust the code
final[c(f1_1), 3] = 5
f1_2 = which(final$Race %in% c(7)) # Adjust the code
final[c(f1_2), 3] = 6
```

```r
# all the variables in our model are categorical variables
# we need to transfer variables from num to factor
final$Diabetes = as.factor(final$Diabetes)
final$Gender = as.factor(final$Gender)
final$Race = as.factor(final$Race)
final$Income = as.factor(final$Income)
final$Sleeping_disorder = as.factor(final$Sleeping_disorder)
final$Cigarettes = as.factor(final$Cigarettes)
final$Alcohol = as.factor(final$Alcohol)

# brief summary of the model and check missing values
str(final)
```

```
## ’data.frame':    8510 obs. of  7 variables:
##  $ Diabetes         : Factor w/ 2 levels "1","2": 2 2 2 2 2 1 2 2 2 2 ...
##  $ Gender           : Factor w/ 2 levels "1","2": 2 1 2 1 1 2 2 1 1 1 ...
##  $ Race             : Factor w/ 6 levels "1","2","3","4",..: 5 3 4 5 6 5 4 5 1 3 ...
##  $ Income           : Factor w/ 4 levels "1","2","3","4": 4 4 1 NA 3 2 1 4 1 2 ...
##  $ Sleeping_disorder: Factor w/ 2 levels "1","2": NA NA 2 2 NA 2 2 1 2 1 ...
##  $ Cigarettes       : Factor w/ 2 levels "1","2": NA NA 2 NA NA NA 1 NA 1 1 ...
##  $ Alcohol          : Factor w/ 5 levels "1","2","3","4",..: NA NA 3 NA NA NA NA 2 5 3 ...
```

```r
summary(final)
```

```
##  Diabetes Gender   Race      Income     Sleeping_disorder Cigarettes
##  1:1023   1:4174   1:1186   1   :1513   1   :1553         1   : 967
##  2:7487   2:4336   2: 704   2   :3730   2   :4321         2   :1295
##                    3:2962   3   :1215   NA’s:2636         NA’s:6248
##                    4:1944   4   :1573
##                    5:1128   NA’s: 479
##                    6: 586
##  Alcohol
##  1   : 289
##  2   : 904
##  3   :1210
##  4   : 960
##  5   :1000
##  NA’s:4147
```

Summary of the final dataset

Parameters	Code:1	Code:2	Code:3	Code:4	Code:5	Code:6
Diabetes	Diabetes & Prediabetes	No				
Gender	Male	Female				
Race	Mexican American	Other Hispanic	Non-Hispanic White	Non-Hispanic Black	Non-Hispanic Asian	Other Race
Annual Household Income	Under $20,000	$20,000 to $64,999	$65,000 to $99,999	$100,000 and over		

Parameters	Code:1	Code:2	Code:3	Code:4	Code:5	Code:6
Trouble Sleeping	Yes	No				
Smoke Cigarettes	Yes	No				
Drink Alcohol	Nearly every day	At least once a week	At least once a month	At least once a year	Never drink	

Best fit model for logistic regression (bestglm)

```r
# Get rid of NA values so that we can run Subset Selection on the model
n1 = which(final$Diabetes %in% c(NA))
n2 = which(final$Gender %in% c(NA))
n3 = which(final$Race %in% c(NA))
n4 = which(final$Income %in% c(NA))
n5 = which(final$Sleeping_disorder %in% c(NA))
n6 = which(final$Cigarettes %in% c(NA))
n7 = which(final$Alcohol %in% c(NA))
# this is the final data set for the model
final_1 = final[-c(n1,n2,n3,n4,n5,n6,n7), ]

# The regsubsets() function only fits linear model. Cannot use leaps package here.
# install.packages(bestglm)
library(bestglm)
# the matrix needs y as the right-most variable
final_2 = final_1[, -1]
final_2 = cbind(final_2, Diabetes = final_1$Diabetes)
best_logit = bestglm(final_2, family = binomial, IC = "AIC", method = "exhaustive")
# best 7 models to be considered
summary(best_logit$BestModel)
```

```
##
## Call:
## glm(formula = y ~ ., family = family, data = Xi, weights = weights)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3708   0.3673   0.5577   0.7032   1.1645
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)          1.775784   0.243235   7.301 2.86e-13 ***
## Gender2              0.370264   0.125892   2.941  0.00327 **
## Sleeping_disorder2   0.595857   0.119921   4.969 6.74e-07 ***
## Cigarettes2         -0.795204   0.128355  -6.195 5.82e-10 ***
## Alcohol2             0.006322   0.257745   0.025  0.98043
## Alcohol3            -0.079647   0.252500  -0.315  0.75243
## Alcohol4            -0.691226   0.254852  -2.712  0.00668 **
## Alcohol5            -0.950109   0.238736  -3.980 6.90e-05 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1930.4  on 1879  degrees of freedom
## Residual deviance: 1794.4  on 1872  degrees of freedom
## AIC: 1810.4
##
## Number of Fisher Scoring iterations: 4
```

```
# the best model with four variables under logistic feature selection method
best_logit$Subsets
```

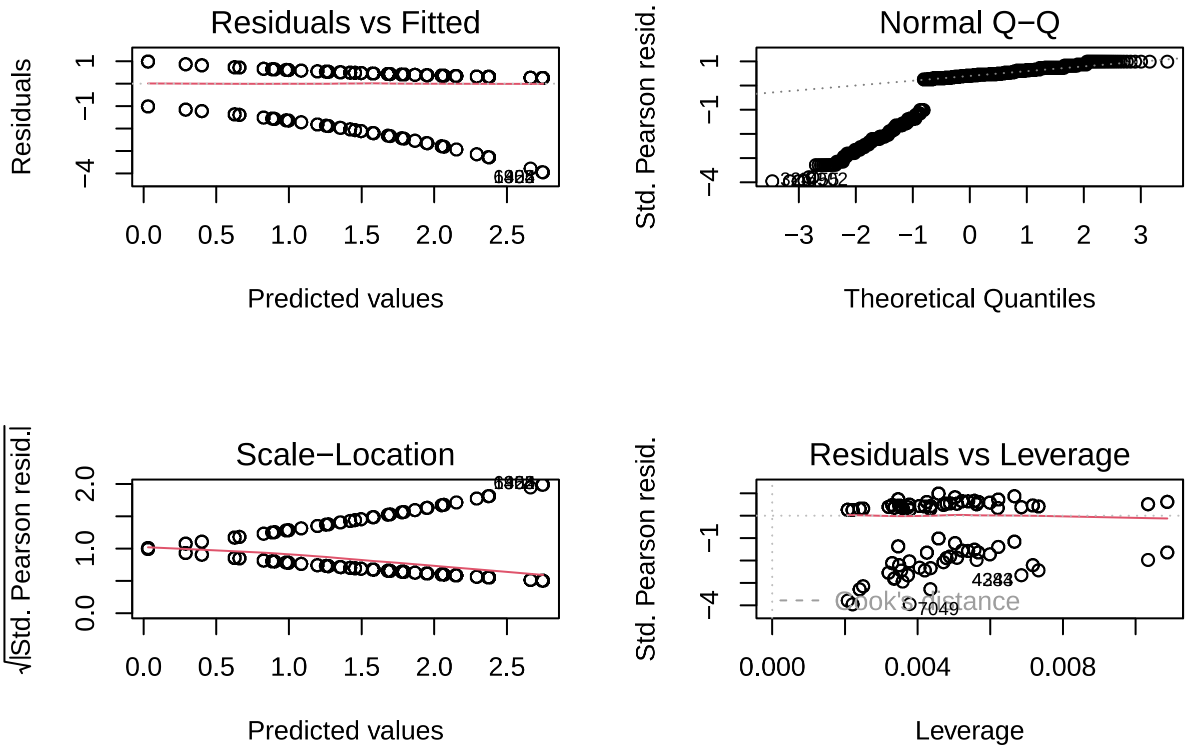
```
##    Intercept Gender  Race Income Sleeping_disorder Cigarettes Alcohol
## 0       TRUE  FALSE FALSE  FALSE             FALSE      FALSE   FALSE
## 1       TRUE  FALSE FALSE  FALSE             FALSE      FALSE    TRUE
## 2       TRUE  FALSE FALSE  FALSE             FALSE       TRUE    TRUE
## 3       TRUE  FALSE FALSE  FALSE              TRUE       TRUE    TRUE
## 4*      TRUE   TRUE FALSE  FALSE              TRUE       TRUE    TRUE
## 5       TRUE   TRUE  TRUE  FALSE              TRUE       TRUE    TRUE
## 6       TRUE   TRUE  TRUE   TRUE              TRUE       TRUE    TRUE
##    logLikelihood      AIC
## 0      -965.1776 1930.355
## 1      -933.9385 1875.877
## 2      -912.4687 1834.937
## 3      -901.6281 1815.256
## 4*     -897.2144 1808.429
## 5      -892.3096 1808.619
## 6      -892.3084 1814.617
```

```
#Another function to run Subset Selection for the logistic model (has the same best fit model)
#library(glmulti)
#   glmulti.logistic.out <-
#     glmulti(Diabetes ~ Gender + Race + Income + Sleeping_disorder +
#                        Cigarettes + Alcohol, data = final_1,
#            level = 1,               # No interaction considered
#            method = "h",            # Exhaustive approach
#            crit = "aic",            # AIC as criteria
#            confsetsize = 5,         # Keep 5 best models
#            plotty = F, report = F,  # No plot or interim reports
#            fitfunction = "glm",     # glm function
#            family = binomial)       # binomial family for logistic regression
#
# Show 5 best models (Use @ instead of $ for an S4 object)
# glmulti.logistic.out@formulas
# summary(glmulti.logistic.out@objects[[1]])
# plot(glmulti.logistic.out@objects[[1]])
```

Interpretation: The best fit model consists of variables gender, sleeping disorders, cigarettes use and alcohol use (Table 2) which means that, with machine learning technique, variables race, and annual household income are not considered in the model. From the summary, the significance levels of alcohol 2 and 3 which are drinking alcohol nearly every day to at least once a week and drinking alcohol at least once a week to at least once a month are bigger than 0.05 which were not considered statistically significant. The log odds of diabetes increase 0.37 from male to female. Log odds of diabetes increase 0.60 from "has sleeping disorders” to “do not have sleeping disorders”. Log odds of diabetes decrease 0.80 from active smoker to “do not smoke”. Log odds of diabetes decrease 0.69 and 0.95 from “drink alcohol at least once a month” to “never drink”. The residual deviance of this model is big which means that this model is not a good fit. However, the difference between Null deviance and Residual deviance is big with 7 degrees of freedom which indicates that the model is satisfied.

Visualization of the best fit logistic regression model

```
par(mfrow = c(2, 2))
plot(best_logit$BestModel)
```



Interpretation: The final table consists of four plots of logistic model (Table 3). From “residuals vs fitted” plot, the upper part looks normal. However, the lower part of this plot shows a pattern of heteroskedasticity which indicates non-constant standard deviations. The “normal Q-Q” plot suggests that the predicted values only starting from predicted values is satisfied. However, logistic regression model does not have normality assumptions. The “residuals vs leverage” plot has no evidence of outliers and none of the observations come close to having both high residual and leverage, since the Cook’s distance dashed curves do not appear on the plot.

Conclusions

Stage 1 indicates that female (8.7%) has lower prevalence of diabetes compared with male (10.6%). And other race has the lowest prevalence of diabetes (8.3%) compared with Mexican American (8.7%), other Hispanic (9.6%), non-Hispanic White (10.4%), non-Hispanic Black (10%), and non-Hispanic Asian (10%).

Stage 2 shows that the best fit model for our hypothesis is that diabetes has relationship with variables gender, sleeping disorders, cigarettes use and alcohol use. The analysis points out that smoking cigarettes less and drinking alcohol less will decrease the log odds of diabetes. However, the only thing that is different from the common sense is that our analysis indicates that people who do not have sleeping disorders has higher log odds of diabetes which need future research.

Limitations

There are many missing values, more than 10% of data, in variables sleeping disorders, cigarettes use, and alcohol use. These missing values may distort the analysis results. The residual deviance of the best fit model and AIC value is large which means that the best model is not a fairly good fit.