Unveiling Obesity dynamics in Allegheny county- A Comprehensive Statistical Analysis of Health and Environmental Factors

Samriddhi Soni

2023-12-04

Installing and loading relevant libraries

```
library(dplyr)
library(tidyverse)
library(ggplot2)
library(gplots)
library(car)
library(corrplot)
library(viridis)
library(jtools)
library(apaTables)
library(ComplexHeatmap)
library(viridis)
library(sf)
```

Load Data

```
data<- read.csv(file="C:/Users/ss6557/Desktop/Semester 3/HUDM 5150-Statistics careers, communica
tion and capstone/data/mergedtablefinal.csv", header = TRUE)

# Extracting relevant variables
data <- data[,c(-11:-22)]
head(data)
```

```
##      CT diabetes conorary_heart_disease binge_drinking obesity sedentary
## 1 10300      6.3                    3.2           24.6    35.2      25.3
## 2 20100      6.8                    4.0           23.9    26.7      16.0
## 3 20300      4.9                    2.8           25.5    26.2      11.8
## 4 30500     22.2                   10.4           12.4    46.4      38.2
## 5 40200      8.7                    4.3           20.2    32.9      26.1
## 6 40500      4.4                    2.8           22.8    29.1      23.9
##   insufficient_sleep health_insurance cholesterol_test_history
## 1               44.5             15.0                     58.1
## 2               35.8              7.3                     73.4
## 3               34.2              5.0                     80.4
## 4               47.7             15.5                     79.7
## 5               42.6             12.7                     61.4
## 6               40.4             12.9                     49.9
##   take_medication_for_hypertention     PM25    Ameds    Dmeds
## 1                             51.9 12.26465 16.58768 19.19431
## 2                             71.8 12.28305 11.68582 12.83525
## 3                             67.6 12.24122 13.27684 11.01695
## 4                             84.3 12.23420 10.82654 12.92200
## 5                             67.1 12.31531 14.17526 16.23711
## 6                             42.7 12.31570 14.88372 12.55814
```

Exploratory data analysis

```
# Correlation Heatmap and correlation plot
cor_mat <- cor(data[,-c(1,5)])
cor_mat
```

```
##                                    diabetes conorary_heart_disease
## diabetes                         1.00000000             0.87986645
## conorary_heart_disease           0.87986645             1.00000000
## binge_drinking                  -0.92671464            -0.85231471
## sedentary                        0.94054318             0.85956805
## insufficient_sleep               0.85721673             0.60475429
## health_insurance                 0.84837972             0.70646441
## cholesterol_test_history        -0.03716442             0.07487846
## take_medication_for_hypertention 0.60485565             0.70891512
## PM25                             0.24339632             0.17972877
## Ameds                           -0.32183423             0.01790541
## Dmeds                            0.04103759             0.25340943
##                                 binge_drinking  sedentary insufficient_sleep
## diabetes                            -0.92671464  0.9405432         0.85721673
## conorary_heart_disease              -0.85231471  0.8595681         0.60475429
## binge_drinking                       1.00000000 -0.8294279        -0.70713149
## sedentary                           -0.82942792  1.0000000         0.91276665
## insufficient_sleep                  -0.70713149  0.9127667         1.00000000
## health_insurance                    -0.69303197  0.9590557         0.94420134
## cholesterol_test_history            -0.15221801 -0.2923847        -0.41752742
## take_medication_for_hypertention    -0.71842040  0.4216402         0.20909307
## PM25                                -0.19260224  0.2971281         0.34173814
## Ameds                                0.30430022 -0.2177231        -0.41257234
## Dmeds                                0.03513604  0.1887963         0.05102663
##                                 health_insurance cholesterol_test_history
## diabetes                               0.8483797              -0.03716442
## conorary_heart_disease                 0.7064644               0.07487846
## binge_drinking                        -0.6930320              -0.15221801
## sedentary                              0.9590557              -0.29238467
## insufficient_sleep                     0.9442013              -0.41752742
## health_insurance                       1.0000000              -0.51077214
## cholesterol_test_history              -0.5107721               1.00000000
## take_medication_for_hypertention       0.1828843               0.66936085
## PM25                                   0.3513653              -0.32871220
## Ameds                                 -0.2678283               0.03059395
## Dmeds                                  0.1927382              -0.30352636
##                                 take_medication_for_hypertention        PM25
## diabetes                                           0.604855652  0.24339632
## conorary_heart_disease                             0.708915119  0.17972877
## binge_drinking                                    -0.718420405 -0.19260224
## sedentary                                          0.421640214  0.29712814
## insufficient_sleep                                 0.209093071  0.34173814
## health_insurance                                   0.182884251  0.35136533
## cholesterol_test_history                           0.669360845 -0.32871220
## take_medication_for_hypertention                   1.000000000 -0.03677360
## PM25                                              -0.036773596  1.00000000
## Ameds                                              0.011357569 -0.02630075
## Dmeds                                             -0.003990822  0.22716776
##                                       Ameds        Dmeds
## diabetes                        -0.32183423  0.041037590
## conorary_heart_disease           0.01790541  0.253409434
## binge_drinking                   0.30430022  0.035136043
```

```
## sedentary                          -0.21772311  0.188796317
## insufficient_sleep                 -0.41257234  0.051026633
## health_insurance                   -0.26782832  0.192738238
## cholesterol_test_history            0.03059395 -0.303526364
## take_medication_for_hypertention    0.01135757 -0.003990822
## PM25                               -0.02630075  0.227167757
## Ameds                               1.00000000  0.748672027
## Dmeds                               0.74867203  1.000000000
```

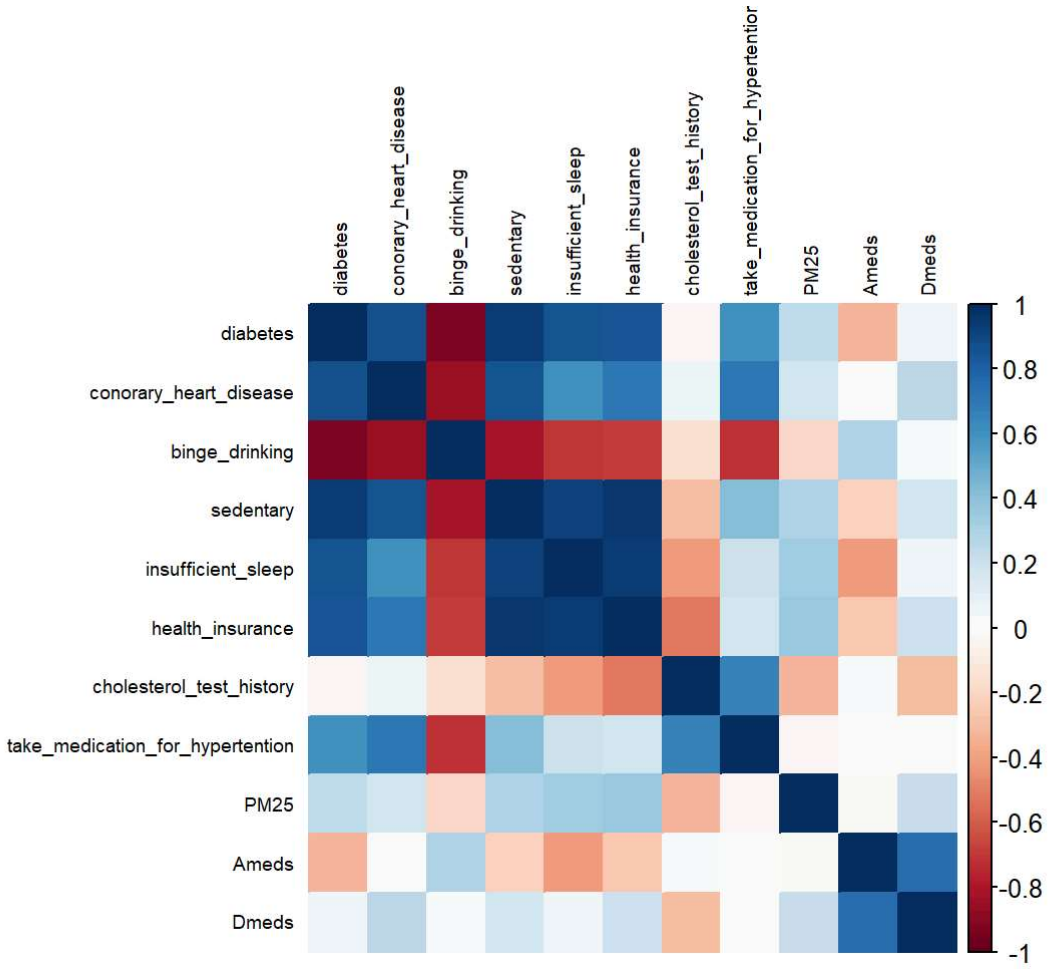
```
apa.cor.table(cor_mat, filename="Correlation matrix.doc", table.number=1)
```

```
##
##
## Table 1
##
## Means, standard deviations, and correlations with confidence intervals
##
##
##   Variable                            M     SD   1             2
##   1. diabetes                         0.38  0.63
##
##   2. conorary_heart_disease           0.40  0.54 .97**
##                                                  [.87, .99]
##
##   3. binge_drinking                   -0.34 0.60 -.99**        -.97**
##                                                  [-1.00, -.94] [-.99, -.90]
##
##   4. sedentary                        0.39  0.62 .98**         .94**
##                                                  [.92, .99]    [.77, .98]
##
##   5. insufficient_sleep               0.31  0.62 .94**         .85**
##                                                  [.79, .99]    [.52, .96]
##
##   6. health_insurance                 0.34  0.62 .93**         .86**
##                                                  [.74, .98]    [.53, .96]
##
##   7. cholesterol_test_history         -0.02 0.47 -.18          -.07
##                                                  [-.70, .47]   [-.64, .55]
##
##   8. take_medication_for_hypertention 0.28  0.47 .69*          .78**
##                                                  [.15, .91]    [.34, .94]
##
##   9. PM25                             0.19  0.35 .42           .33
##                                                  [-.24, .81]   [-.34, .78]
##
##   10. Ameds                           0.08  0.44 -.71*         -.53
##                                                  [-.92, -.19]  [-.86, .10]
##
##   11. Dmeds                           0.22  0.36 -.19          -.06
##                                                  [-.71, .46]   [-.63, .56]
##
##   3            4            5            6            7            8
##
##
##
##
##
##
##
##
##
##   -.93**
##   [-.98, -.76]
##
```

```
##   -.88**       .98**
##   [-.97, -.58] [.92, .99]
##
##   -.85**       .98**        .99**
##   [-.96, -.52] [.93, 1.00]  [.97, 1.00]
##
##   .02          -.37         -.48         -.53
##   [-.59, .61]  [-.79, .30]  [-.84, .17]  [-.86, .10]
##
##   -.80**       .54          .41          .37          .57
##   [-.95, -.38] [-.08, .86]  [-.25, .81]  [-.29, .80]  [-.05, .87]
##
##   -.34         .51          .58          .60          -.65*        -.11
##   [-.78, .33]  [-.12, .85]  [-.03, .88]  [-.00, .88]  [-.90, -.09] [-.67, .52]
##
##   .69*         -.66*        -.73*        -.65*        .10          -.39
##   [.16, .91]   [-.90, -.09] [-.93, -.23] [-.90, -.08] [-.53, .66]  [-.80, .27]
##
##   .25          -.05         -.10         .01          -.45         -.34
##   [-.41, .74]  [-.63, .57]  [-.66, .53]  [-.59, .60]  [-.83, .20]  [-.78, .33]
##
##   9           10
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##   -.31
##   [-.77, .35]
##
##   .21         .73*
```

```
##   [-.44, .72] [.24, .93]
##
##
## Note. M and SD are used to represent mean and standard deviation, respectively.
## Values in square brackets indicate the 95% confidence interval.
## The confidence interval is a plausible range of population correlations
## that could have caused the sample correlation (Cumming, 2014).
##  * indicates p < .05. ** indicates p < .01.
##
```

```
# Correlation plot
corrplot(cor_mat,method="color",tl.col = "black", tl.cex = 0.6)
```



Data transformation based on EDA

```
# combining variables with high level of correlation
data$ob_sed <- (data$obesity + data$sedentary) / 2
data$am_dmeds <- (data$Ameds + data$Dmeds) / 2
```

OLS multiple linear regression

```
ols_model <- lm(data[, 2] ~ ., data = data[,c(-1,-2,-5,-6,-7,-8,-12,-13)])
summary(ols_model)
```

```
##
## Call:
## lm(formula = data[, 2] ~ ., data = data[, c(-1, -2, -5, -6, -7,
##     -8, -12, -13)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7917 -0.3600 -0.0637  0.2445  3.6384
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     -3.37703    3.06562  -1.102   0.2713
## conorary_heart_disease           0.36431    0.05069   7.187 3.55e-12 ***
## binge_drinking                  -0.31693    0.03473  -9.126  < 2e-16 ***
## cholesterol_test_history         0.04609    0.02031   2.269   0.0238 *
## take_medication_for_hypertention  0.02591    0.02567   1.009   0.3134
## PM25                             0.23781    0.18325   1.298   0.1952
## ob_sed                           0.40503    0.01613  25.113  < 2e-16 ***
## am_dmeds                        -0.14812    0.02121  -6.983 1.30e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6897 on 379 degrees of freedom
## Multiple R-squared:  0.9737, Adjusted R-squared:  0.9732
## F-statistic:  2003 on 7 and 379 DF,  p-value: < 2.2e-16
```

```
apa.reg.table(ols_model, filename="Regression.doc", table.number=2)
```

```
##
##
## Table 2
##
## Regression results using data[, 2] as the criterion
##
##
##                          Predictor       b       b_95%_CI  beta    beta_95%_CI
##                        (Intercept)   -3.38  [-9.40, 2.65]
##            conorary_heart_disease  0.36**   [0.26, 0.46]  0.17   [0.13, 0.22]
##                    binge_drinking -0.32** [-0.39, -0.25] -0.22 [-0.27, -0.18]
##          cholesterol_test_history   0.05*   [0.01, 0.09]  0.05   [0.01, 0.09]
##  take_medication_for_hypertention    0.03  [-0.02, 0.08]  0.03  [-0.03, 0.08]
##                               PM25    0.24  [-0.12, 0.60]  0.01  [-0.01, 0.03]
##                             ob_sed  0.41**   [0.37, 0.44]  0.61   [0.57, 0.66]
##                           am_dmeds -0.15** [-0.19, -0.11] -0.08 [-0.10, -0.06]
##
##
##
##  sr2  sr2_95%_CI      r             Fit
##
##  .00  [.00, .01]  .88**
##  .01  [.00, .01] -.93**
##  .00 [-.00, .00]   -.04
##  .00 [-.00, .00]  .60**
##  .00 [-.00, .00]  .24**
##  .04  [.03, .05]  .95**
##  .00  [.00, .01] -.16**
##                              R2 = .974**
##                           95% CI[.97,.98]
##
##
## Note. A significant b-weight indicates the beta-weight and semi-partial correlation are also
significant.
## b represents unstandardized regression weights. beta indicates the standardized regression we
ights.
## sr2 represents the semi-partial correlation squared. r represents the zero-order correlation.
## Square brackets are used to enclose the lower and upper limits of a confidence interval.
## * indicates p < .05. ** indicates p < .01.
##
```

```
# Extract coefficients
coefs <- summary(ols_model)$coefficients

# Get only significant predictors (ignoring the intercept)
significant_vars <- rownames(coefs)[which(coefs[, 4] < 0.05 & rownames(coefs) != "(Intercept)")]
significant_vars
```

```
## [1] "conorary_heart_disease"   "binge_drinking"
## [3] "cholesterol_test_history" "ob_sed"
## [5] "am_dmeds"
```

Detecting outliers

```
# Install and load the 'car' package
library(car)
library(latticeExtra)
```

```
## Warning: package 'latticeExtra' was built under R version 4.2.3
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'latticeExtra'
```

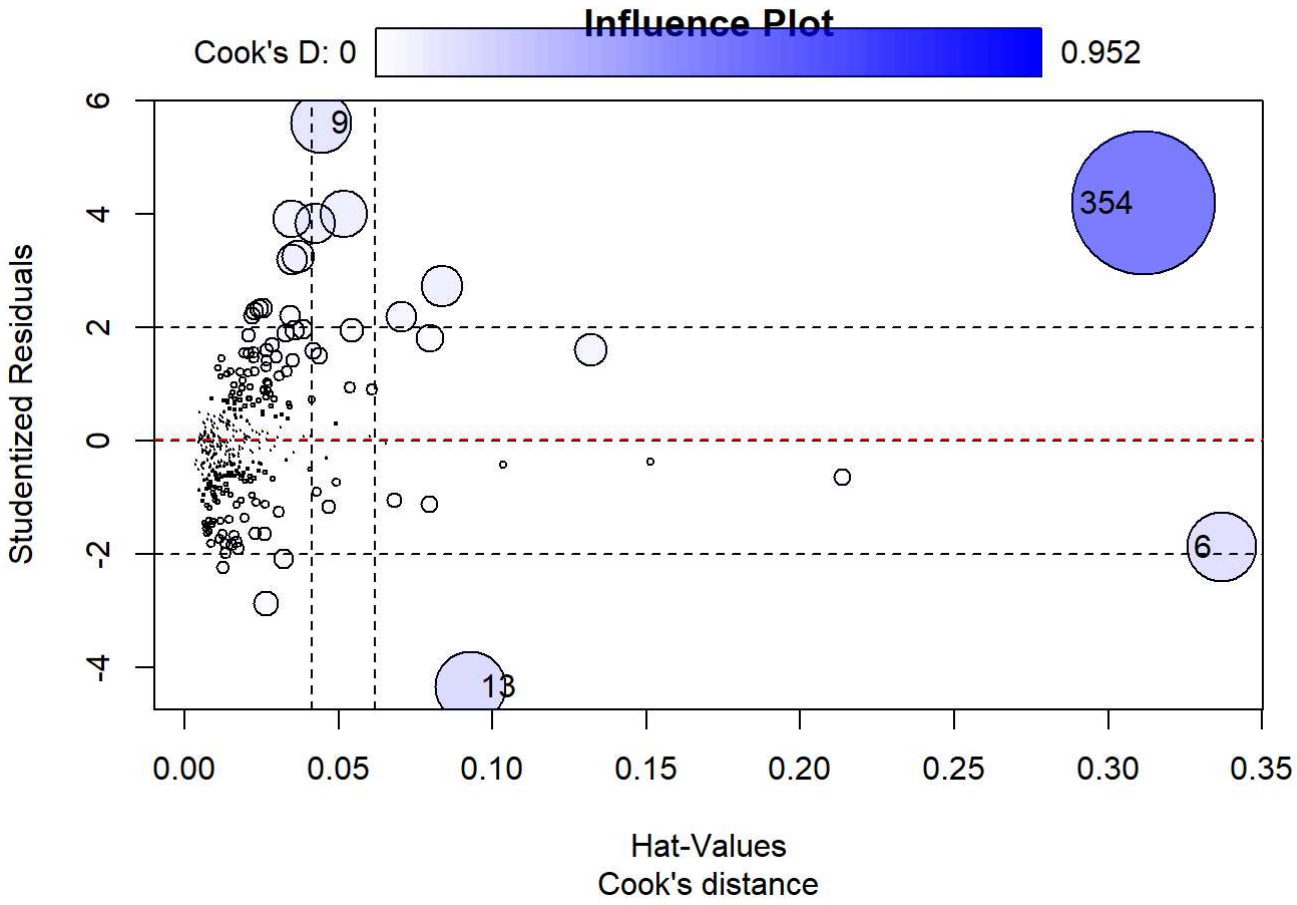
```
## The following object is masked from 'package:ComplexHeatmap':
##
##     dendrogramGrob
```

```
## The following object is masked from 'package:ggplot2':
##
##     layer
```

```
# Create a linear regression model
model <- lm(mpg ~ wt + hp + qsec, data = mtcars)

# Create an influence plot
influencePlot(ols_model, main="Influence Plot", sub="Cook's distance")
```

```
##       StudRes        Hat     CookD
## 6   -1.881010 0.33690112 0.2232120
## 9    5.609079 0.04434212 0.1689013
## 13  -4.348903 0.09279967 0.2309174
## 354  4.190901 0.31146743 0.9515593
```

```
# Add a reference line for Cook's distance threshold (e.g., 4/n)
abline(h = 4/(length(ols_model$residuals)), col="red", lty=2)
```




```
# Get Cook's distance values
cooksd <- cooks.distance(ols_model)

# Set a threshold for Cook's distance (you can adjust this)
threshold <- 4 / length(cooksd)

# Identify influential observations
influential_observations <- which(cooksd > threshold)

# Print or use influential_observations as needed
print(influential_observations)
```

```
##   2   3   4   6   7   9  12  13  17  18  39  41  42  43  44  45  46  49  63  84
##   2   3   4   6   7   9  12  13  17  18  39  41  42  43  44  45  46  49  63  84
##  88  98 185 352 354 362 363 368 369
##  88  98 185 352 354 362 363 368 369
```

Removing the four influential points

```
# Extracting relevant variables
x <- c(2  , 3 ,  4  , 6 ,  7  , 9 , 12  ,13 , 17  ,18 , 39  ,41  ,42  ,43 , 44  ,45 , 46 , 49,
63  ,84 , 88  ,98 ,185 ,352 ,354 ,363 ,368 ,369)
data_new <- data[-x,]
head(data_new)
```

```
##       CT diabetes conorary_heart_disease binge_drinking obesity sedentary
## 1  10300      6.3                    3.2           24.6    35.2      25.3
## 5  40200      8.7                    4.3           20.2    32.9      26.1
## 8  40900     10.5                    6.3           20.7    34.9      26.8
## 10 50600     17.3                    9.0           13.0    39.3      30.4
## 11 50900     24.8                   11.7           10.9    52.7      46.3
## 14 60300      8.3                    5.1           22.0    31.4      19.0
##    insufficient_sleep health_insurance cholesterol_test_history
## 1                44.5             15.0                      58.1
## 5                42.6             12.7                      61.4
## 8                41.9             12.5                      64.3
## 10               42.9             11.5                      85.0
## 11               53.0             20.4                      74.8
## 14               37.5              8.0                      77.7
##    take_medication_for_hypertention     PM25    Ameds    Dmeds ob_sed am_dmeds
## 1                              51.9 12.26465 16.58768 19.19431  30.25 17.89100
## 5                              67.1 12.31531 14.17526 16.23711  29.50 15.20619
## 8                              70.3 12.33776 18.36115 17.75417  30.85 18.05766
## 10                             84.5 12.22199 11.67315 11.41375  34.85 11.54345
## 11                             81.5 12.24335 12.88981 15.59252  49.50 14.24116
## 14                             74.4 12.17729 18.40844 16.58677  25.20 17.49760
```

Fitting the new OLS regression model

```
# OLS Multiple Linear regression Model

new_model <- lm(data_new[, 2] ~ ., data = data_new[,c(-1,-2,-5,-6,-7,-8, -10,-12,-13)])
summary(new_model)
```

```
##
## Call:
## lm(formula = data_new[, 2] ~ ., data = data_new[, c(-1, -2, -5,
##     -6, -7, -8, -10, -12, -13)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.3586 -0.2033 -0.0367  0.1753  3.4730
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -2.40805    2.32315  -1.037  0.30066
## conorary_heart_disease    0.11099    0.03920   2.831  0.00490 **
## binge_drinking           -0.38081    0.02438 -15.620  < 2e-16 ***
## cholesterol_test_history  0.08207    0.01118   7.343 1.46e-12 ***
## PM25                      0.15009    0.12340   1.216  0.22467
## ob_sed                    0.43123    0.01270  33.951  < 2e-16 ***
## am_dmeds                 -0.06100    0.01617  -3.772  0.00019 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4577 on 352 degrees of freedom
## Multiple R-squared:  0.9804, Adjusted R-squared:   0.98
## F-statistic:  2929 on 6 and 352 DF,  p-value: < 2.2e-16
```

```
apa.reg.table(ols_model, filename="Regression.doc", table.number=2)
```

```
##
##
## Table 2
##
## Regression results using data[, 2] as the criterion
##
##
##                          Predictor       b       b_95%_CI  beta    beta_95%_CI
##                        (Intercept)   -3.38  [-9.40, 2.65]
##            conorary_heart_disease  0.36**   [0.26, 0.46]  0.17   [0.13, 0.22]
##                    binge_drinking -0.32** [-0.39, -0.25] -0.22 [-0.27, -0.18]
##          cholesterol_test_history   0.05*   [0.01, 0.09]  0.05   [0.01, 0.09]
##  take_medication_for_hypertention    0.03  [-0.02, 0.08]  0.03  [-0.03, 0.08]
##                               PM25    0.24  [-0.12, 0.60]  0.01  [-0.01, 0.03]
##                             ob_sed  0.41**   [0.37, 0.44]  0.61   [0.57, 0.66]
##                           am_dmeds -0.15** [-0.19, -0.11] -0.08 [-0.10, -0.06]
##
##
##
##  sr2  sr2_95%_CI      r             Fit
##
##  .00  [.00, .01]  .88**
##  .01  [.00, .01] -.93**
##  .00 [-.00, .00]   -.04
##  .00 [-.00, .00]  .60**
##  .00 [-.00, .00]  .24**
##  .04  [.03, .05]  .95**
##  .00  [.00, .01] -.16**
##                              R2 = .974**
##                           95% CI[.97,.98]
##
##
## Note. A significant b-weight indicates the beta-weight and semi-partial correlation are also
significant.
## b represents unstandardized regression weights. beta indicates the standardized regression we
ights.
## sr2 represents the semi-partial correlation squared. r represents the zero-order correlation.
## Square brackets are used to enclose the lower and upper limits of a confidence interval.
## * indicates p < .05. ** indicates p < .01.
##
```

```
# Extract coefficients
coefs <- summary(new_model)$coefficients

# Get only significant predictors (ignoring the intercept)
significant_vars <- rownames(coefs)[which(coefs[, 4] < 0.05 & rownames(coefs) != "(Intercept)")]
significant_vars
```

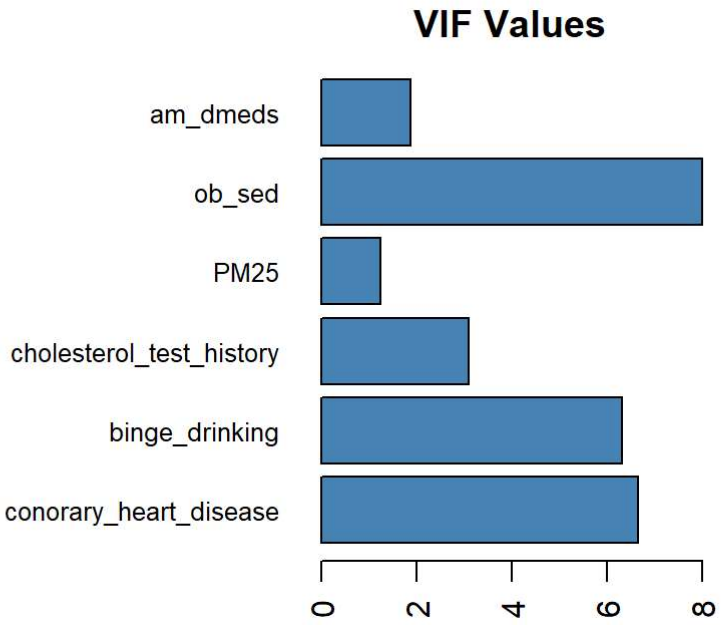
```
## [1] "conorary_heart_disease"   "binge_drinking"
## [3] "cholesterol_test_history" "ob_sed"
## [5] "am_dmeds"
```

Testing for multicollinearity

```
# Test for multicollinearity
# Checking the VIF
vif_values <- vif(new_model)
vif_values
```

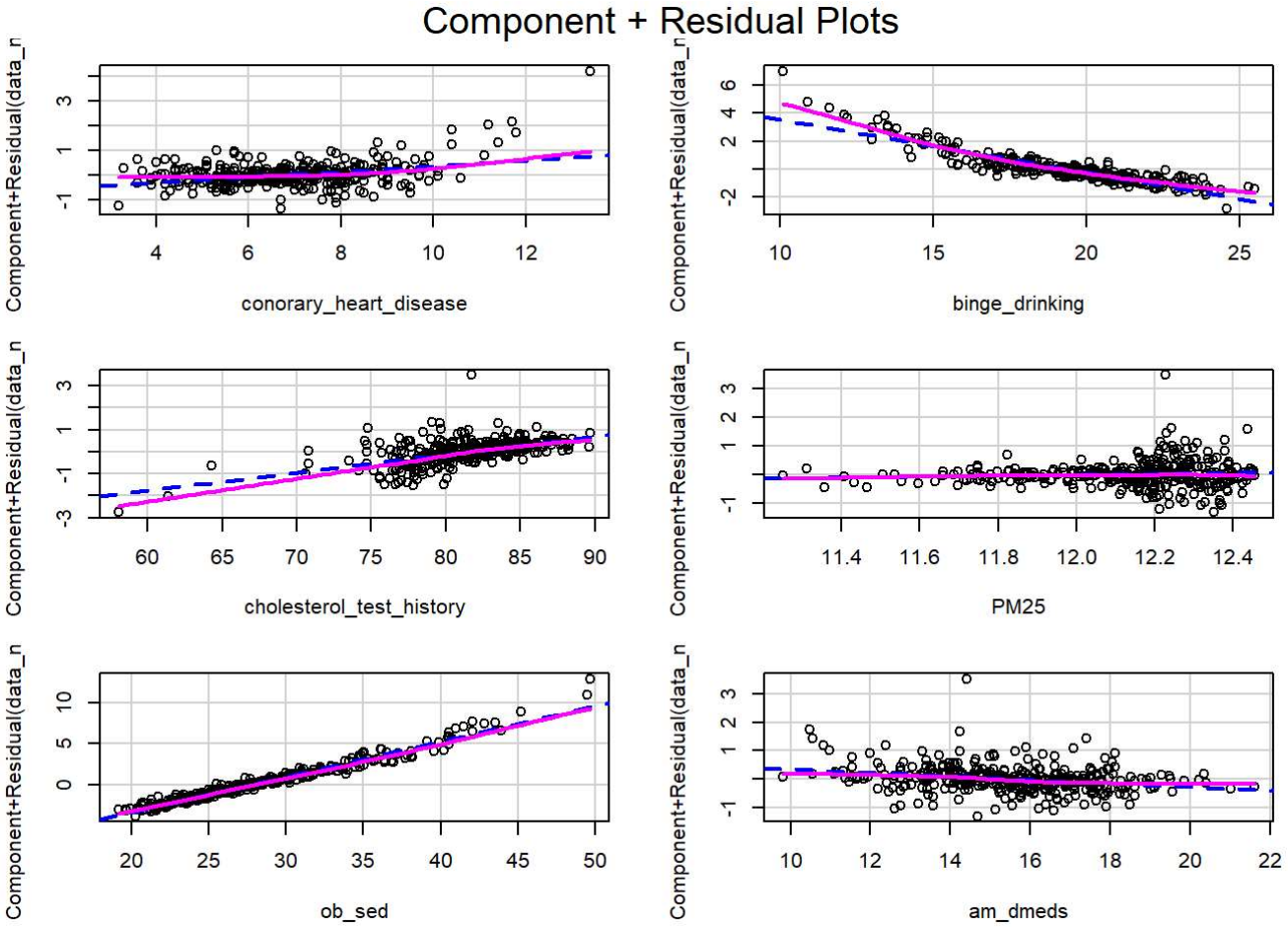
```
##   conorary_heart_disease          binge_drinking cholesterol_test_history
##                 6.668306                6.318014                3.092620
##                     PM25                  ob_sed                 am_dmeds
##                 1.230955                8.020781                1.877478
```

```
par(mar = c(10, 15, 2, 10))
barplot(vif_values, main = "VIF Values", horiz = TRUE, col = "steelblue",cex.names = 0.8, las=2)
abline(v = 10, lwd = 3, lty = 2)
```

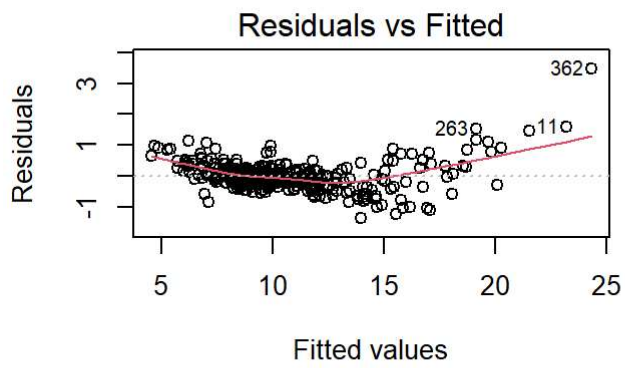


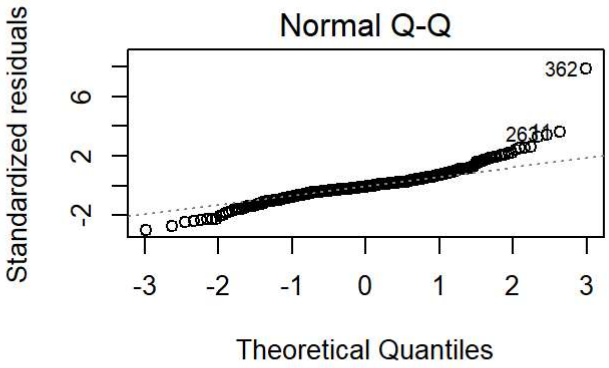
Diagnostic plots for testing the theoretical assumptions

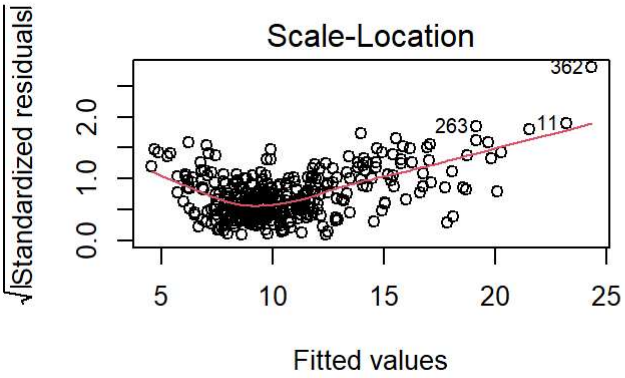
```
crPlots(new_model)
```

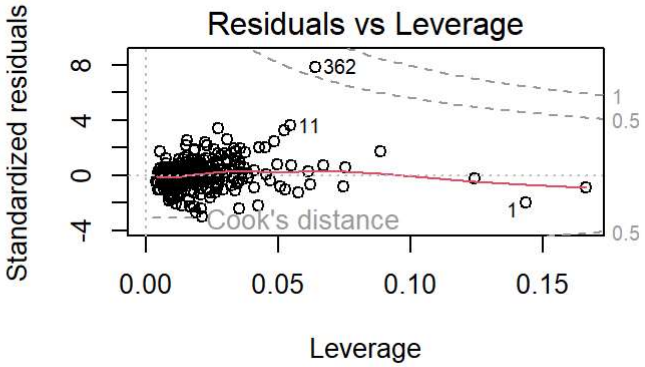



```
par(mfrow=c(2,2))
plot(new_model)
```



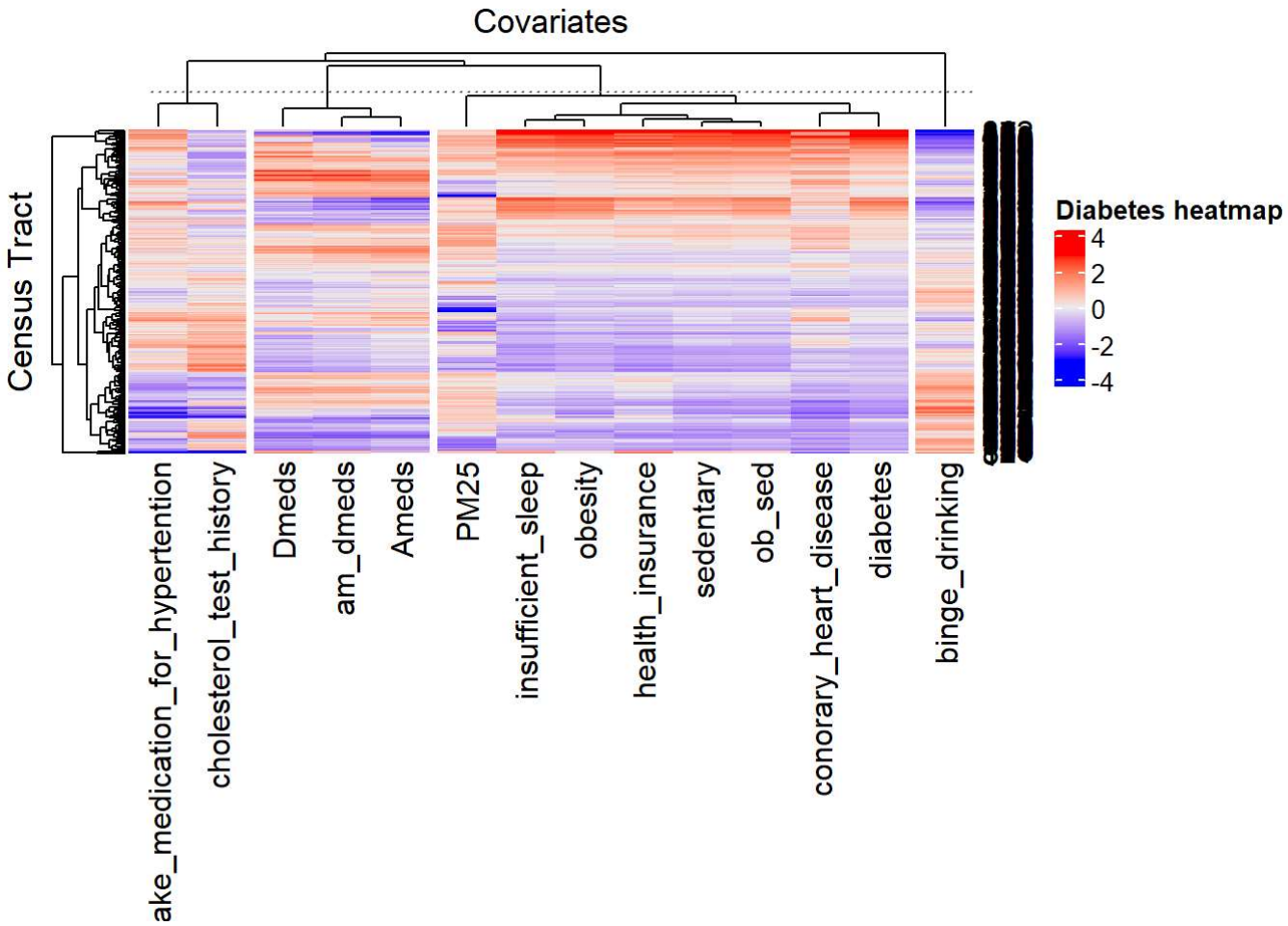






Heatmap Clustering

```
#Heat map clustering
data_scale <- scale(data_new[,-1])
Heatmap(data_scale, name="Diabetes heatmap",row_title = "Census Tract",column_title = "Covariate
s", column_km = 4)
```



Regression trees and Random forest

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.2.3
```

```
## randomForest 4.7-1.1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.2.3
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(pdp)
```

```
## Warning: package 'pdp' was built under R version 4.2.3
```

```
##
## Attaching package: 'pdp'
```

```
## The following object is masked from 'package:purrr':
##
##     partial
```

```
library(rpart)
```

```
## Warning: package 'rpart' was built under R version 4.2.3
```

```
library(rpart.plot)
```

```
## Warning: package 'rpart.plot' was built under R version 4.2.3
```

```
library(modelr)
```

```
## Warning: package 'modelr' was built under R version 4.2.3
```

```
data <- data_new[,c(-1,-5,-6,-12,-13)]
head(data)
```

```
##    diabetes conorary_heart_disease binge_drinking insufficient_sleep
## 1       6.3                   3.2           24.6                44.5
## 5       8.7                   4.3           20.2                42.6
## 8      10.5                   6.3           20.7                41.9
## 10     17.3                   9.0           13.0                42.9
## 11     24.8                  11.7           10.9                53.0
## 14      8.3                   5.1           22.0                37.5
##    health_insurance cholesterol_test_history take_medication_for_hypertention
## 1              15.0                     58.1                              51.9
## 5              12.7                     61.4                              67.1
## 8              12.5                     64.3                              70.3
## 10             11.5                     85.0                              84.5
## 11             20.4                     74.8                              81.5
## 14              8.0                     77.7                              74.4
##        PM25 ob_sed am_dmeds
## 1  12.26465  30.25 17.89100
## 5  12.31531  29.50 15.20619
## 8  12.33776  30.85 18.05766
## 10 12.22199  34.85 11.54345
## 11 12.24335  49.50 14.24116
## 14 12.17729  25.20 17.49760
```

Creating training and testing data set

```
set.seed(123)

# Splitting the dataset into training and test.
trainIndex <- sample(1:nrow(data), size = 0.7*nrow(data))

trainData <- data[trainIndex, ]
testData <- data[-trainIndex, ]
```

Regression tree

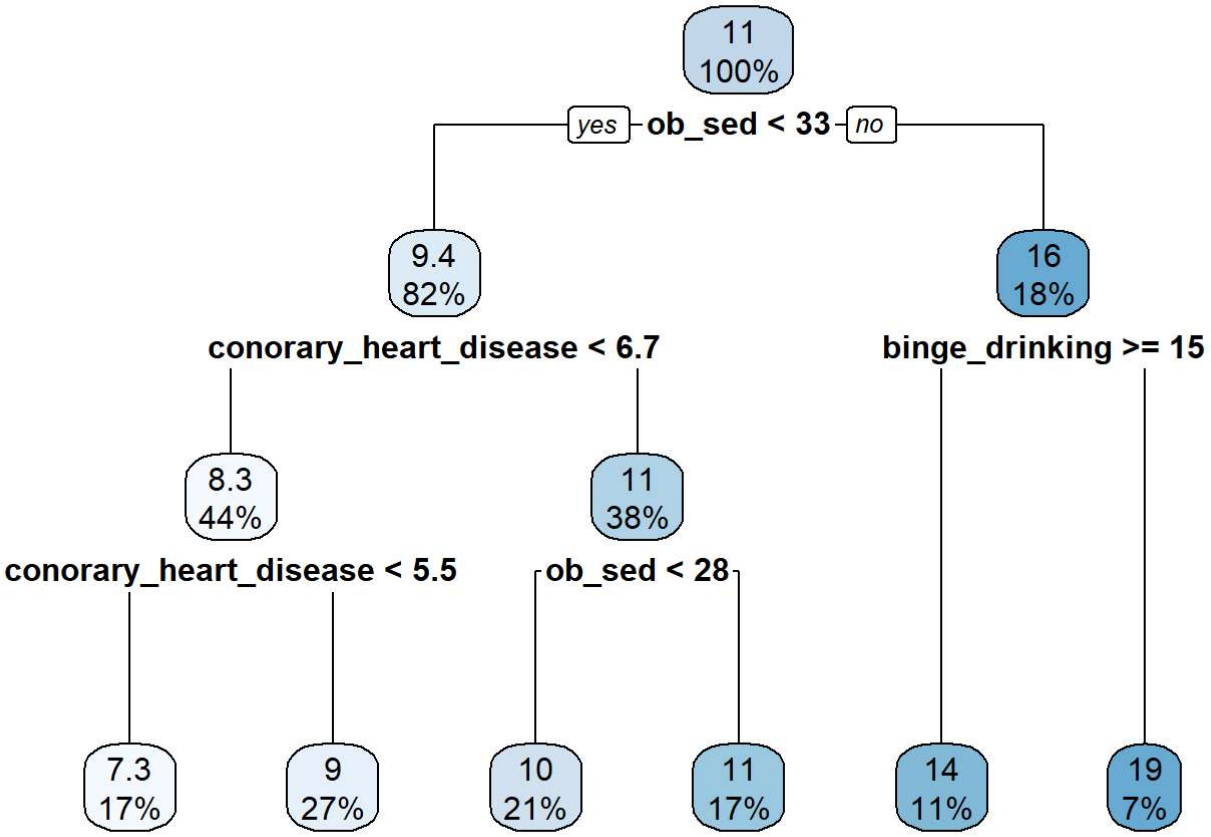
```
#Since the Random Forest model does not observe individual dendograms, I additionally added a de
cision tree model for need.

# Training a regression tree model on the training data.
rt_model <- rpart(diabetes ~ ., data = trainData, method = "anova")

# Making predictions on the test data using the decision tree model.
prediction <- predict(rt_model, testData)

# Calculating the MSE for the decision tree predictions.
mse <- mean((testData$diabetes - prediction)^2)

# Calculating the Root Mean Squared Error (RMSE) from the MSE.
rmse <- sqrt(mse)

# Printing the MSE and RMSE.
print(paste("MSE:", mse))
```

```
## [1] "MSE: 1.77697668981073"
```

```
print(paste("RMSE:", rmse))
```

```
## [1] "RMSE: 1.3330328914962"
```

```
# Plotting the decision tree using rpart.plot.
rpart.plot(rt_model)
```




```
# Calculating variable importance for the decision tree model.
varImp(rt_model)
```

```
##                                  Overall
## binge_drinking                  2.167346
## conorary_heart_disease          2.463068
## health_insurance                1.276974
## insufficient_sleep              1.713475
## ob_sed                          2.721522
## take_medication_for_hypertention 1.265007
## cholesterol_test_history        0.000000
## PM25                            0.000000
## am_dmeds                        0.000000
```

Random forest

```
# Training a random forest regression model on the training data.
rf_regression_model <- randomForest(diabetes ~ ., data=trainData, method="anova")

print(rf_regression_model)
```

```
##
## Call:
##  randomForest(formula = diabetes ~ ., data = trainData, method = "anova")
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##           Mean of squared residuals: 0.2721617
##                     % Var explained: 97.31
```

```
# Making predictions on the test data.
predictions <- predict(rf_regression_model, newdata=testData)

# Calculating the Mean Squared Error (MSE) for the predictions.
mse <- mean((testData$diabetes - predictions)^2)

print(paste("MSE:", mse))
```

```
## [1] "MSE: 0.386356928806074"
```

Variable importance for the random forest model

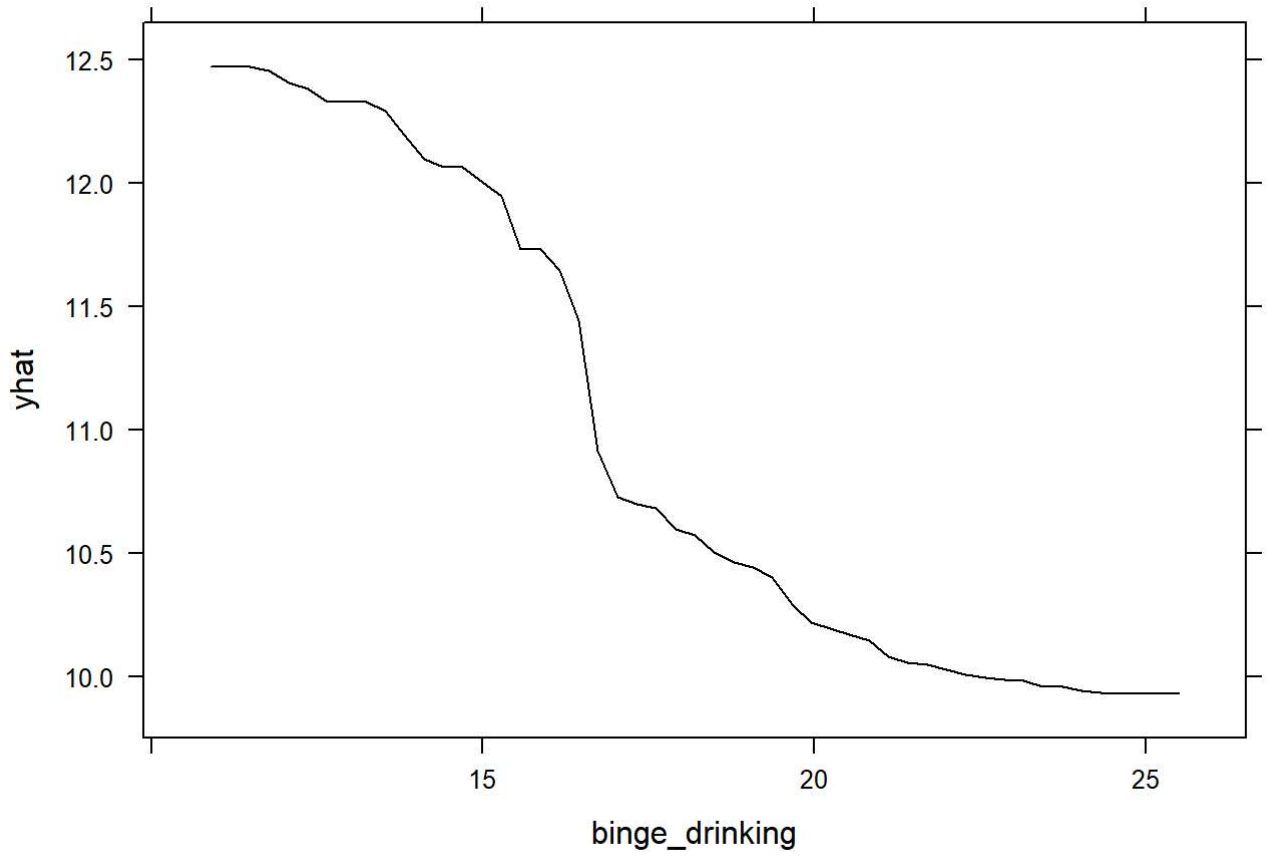
```
 #Calculating the importance of each feature in the random forest model.
importance(rf_regression_model)
```

```
##                                  IncNodePurity
## conorary_heart_disease              304.453972
## binge_drinking                      536.926552
## insufficient_sleep                  389.187517
## health_insurance                    274.577328
## cholesterol_test_history             28.373491
## take_medication_for_hypertention    120.180107
## PM25                                  8.751946
## ob_sed                              824.525574
## am_dmeds                             14.753706
```

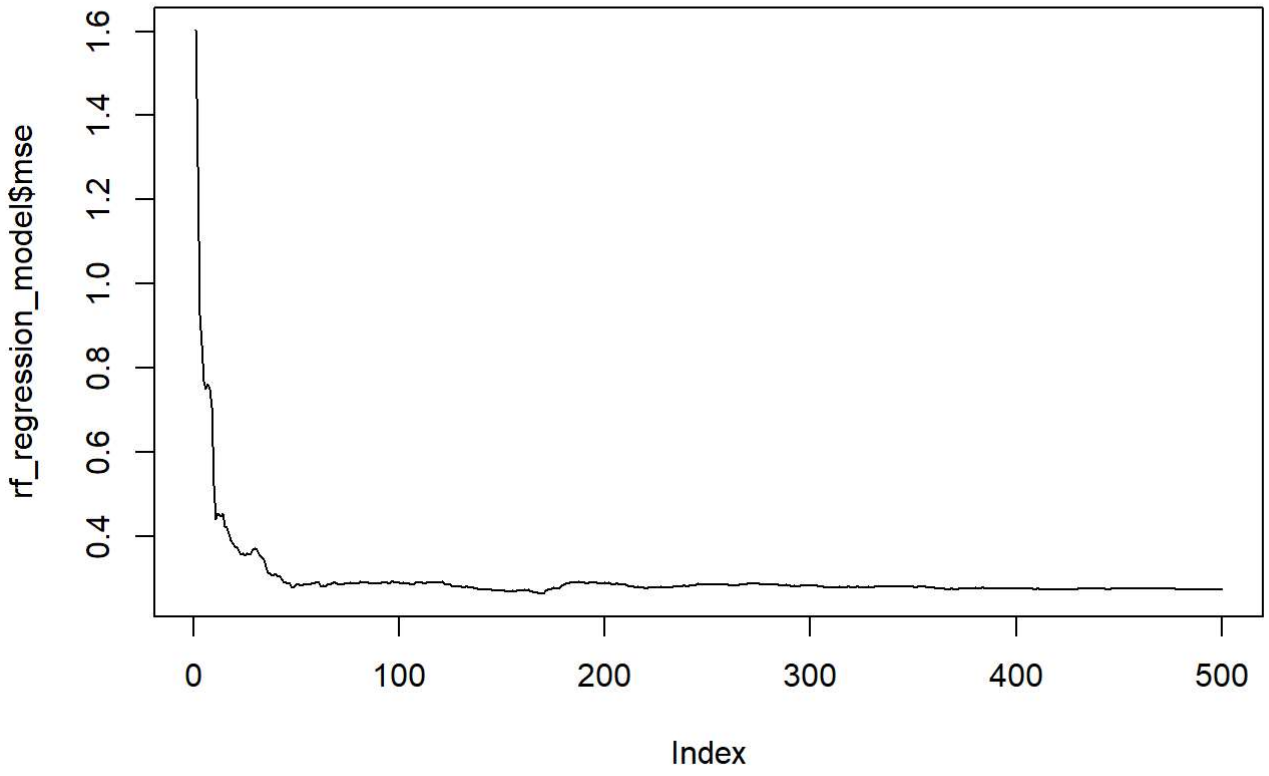
```
# Plotting the importance of each feature.
importance_data <- importance(rf_regression_model)
importance_df <- data.frame(Variable=rownames(importance_data), Importance=importance_data[,1])
importance_df <- importance_df[order(-importance_df$Importance), ]
print(importance_df)
```

```
##                                                          Variable Importance
## ob_sed                                                     ob_sed 824.525574
## binge_drinking                                     binge_drinking 536.926552
## insufficient_sleep                             insufficient_sleep 389.187517
## conorary_heart_disease                     conorary_heart_disease 304.453972
## health_insurance                                   health_insurance 274.577328
## take_medication_for_hypertention take_medication_for_hypertention 120.180107
## cholesterol_test_history                 cholesterol_test_history  28.373491
## am_dmeds                                                 am_dmeds  14.753706
## PM25                                                         PM25   8.751946
```

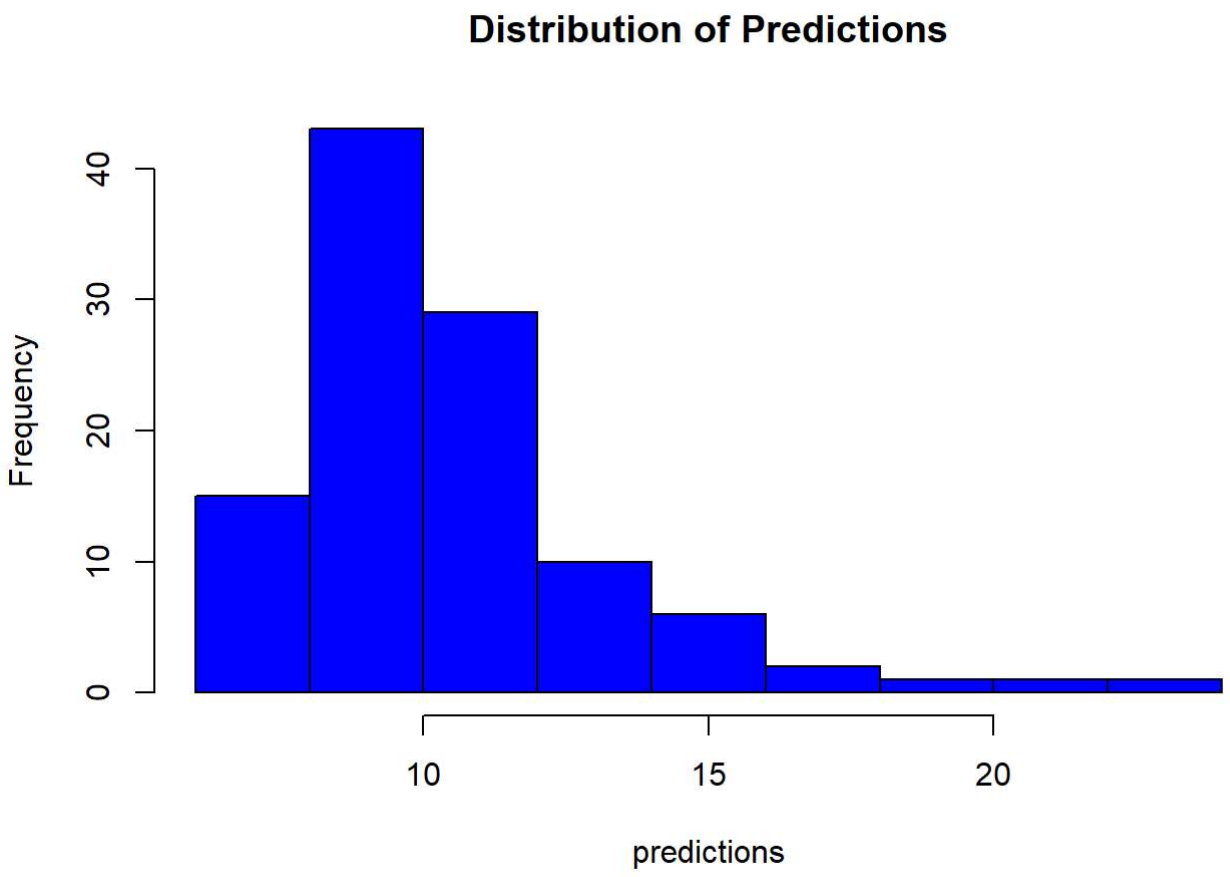
```
# Creating a partial dependence plot for the 'binge_drinking' variable.
pd <- partial(rf_regression_model, pred.var = "binge_drinking")
plotPartial(pd)
```



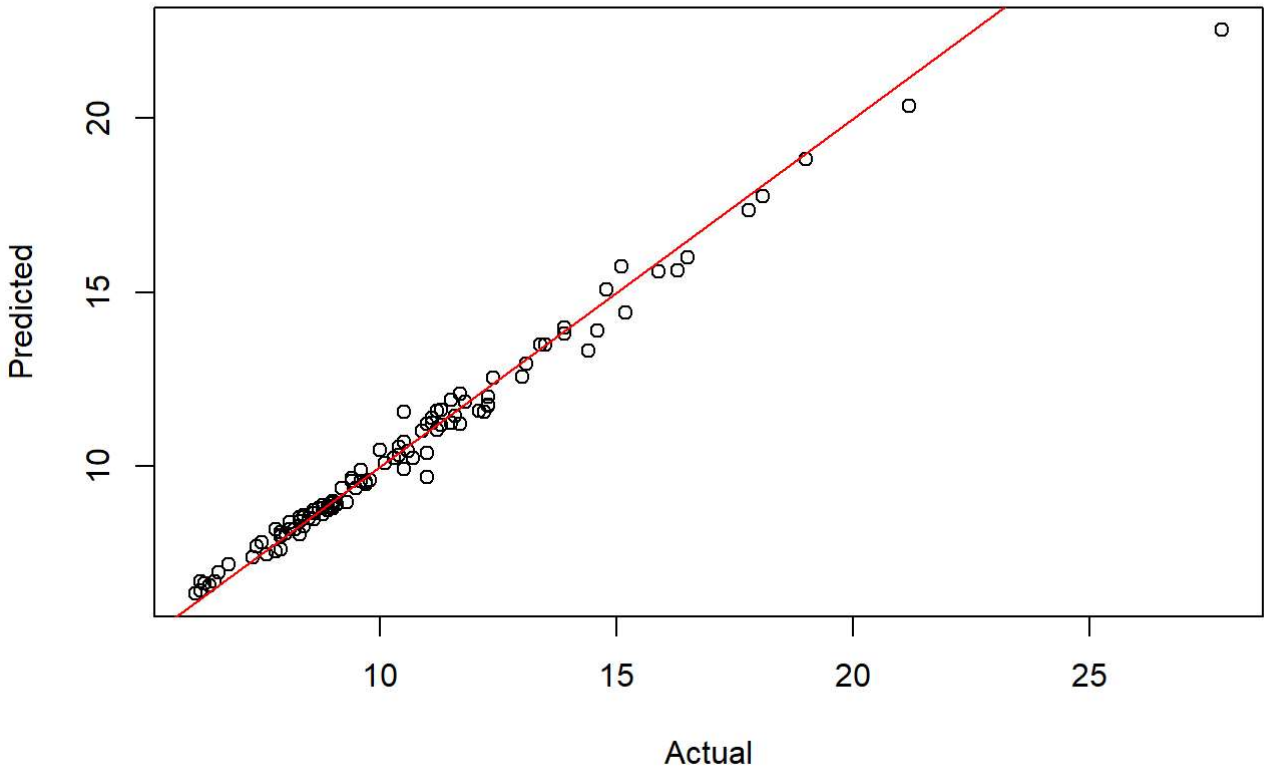

```
# Plotting the MSE of the random forest model as the number of trees increases.
plot(rf_regression_model$mse, type="l")
```



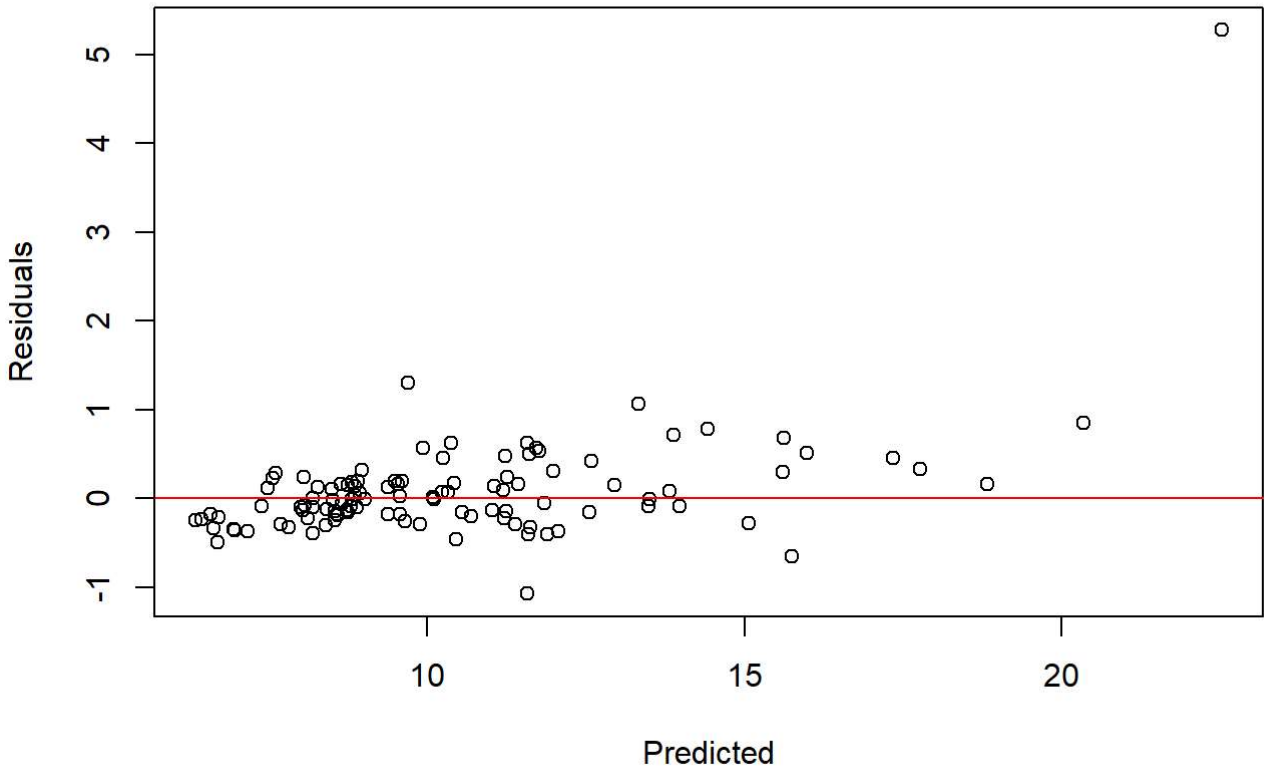

```
# Plotting a histogram of the predicted values.
hist(predictions, col='blue', main='Distribution of Predictions')
```




```
#Plotting scatter plots of predicted versus actual values
plot(testData$diabetes, predictions, xlab = "Actual", ylab = "Predicted")
abline(0, 1, col = "red")
```




```
#Plotting scatter plots of residuals versus predicted values
residuals <- testData$diabetes - predictions
plot(predictions, residuals, xlab = "Predicted", ylab = "Residuals")
abline(h = 0, col = "red")
```


Residuals
-1
0
1
2
3
4
5
10
15
20
Predicted