yifan_final

Yifan Duan

2022-12-06

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift
```

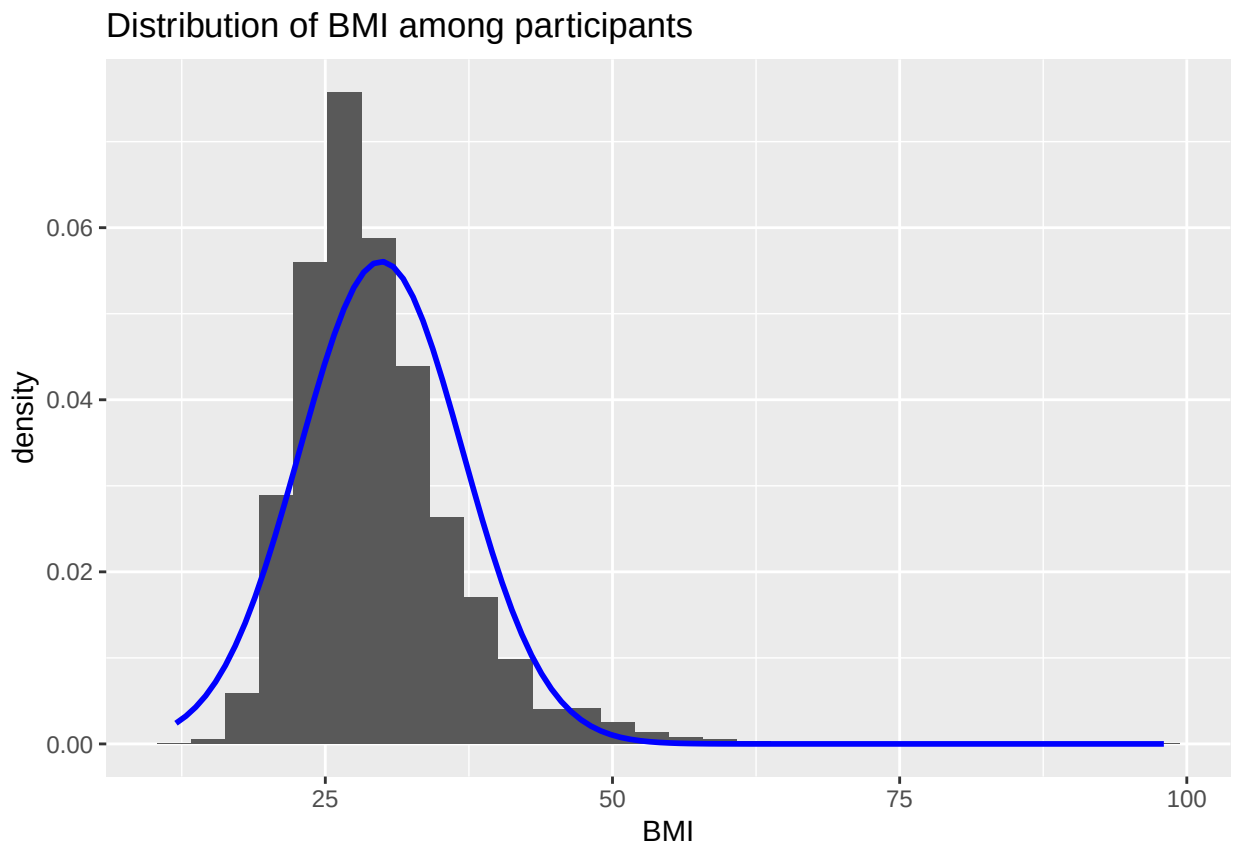
```
df <- read.csv("diabetes_5050.csv", sep = ",", header = TRUE)
```

```
df |>
  ggplot() +
  geom_histogram(aes(x = BMI, y = ..density..)) +
  labs(x = "BMI", title = "Distribution of BMI among participants") +
  stat_function(fun = dnorm, args = list(mean = mean(df$BMI), sd = sd(df$BMI)), lwd = 1, col = "blue")
```

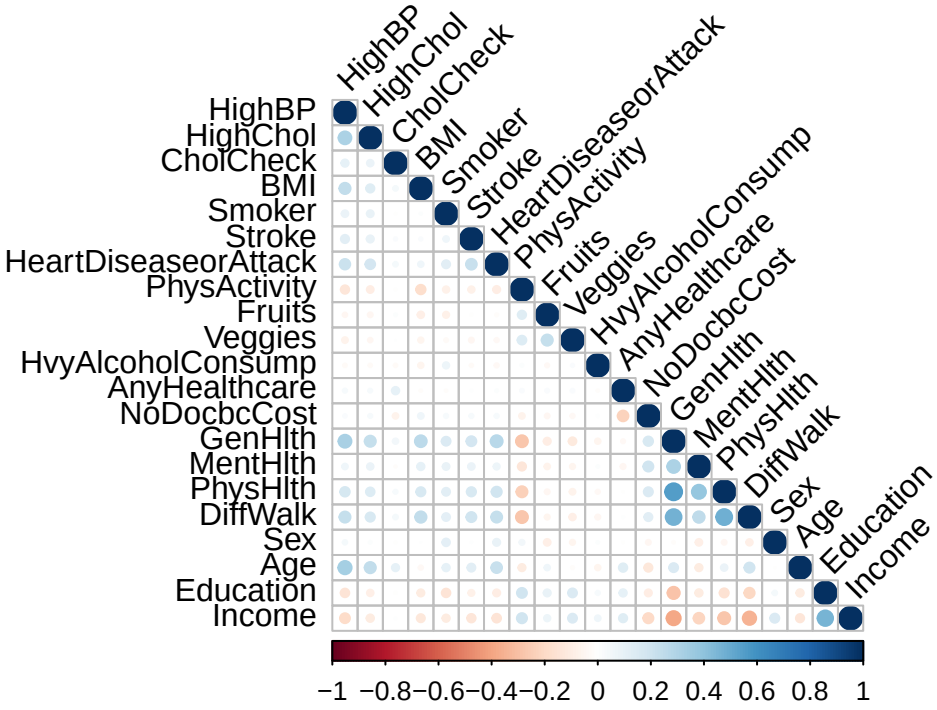
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
df_cor = cor(df[, c(2:22)], method = "pearson")
corrplot(df_cor, method = "circle", type = "lower", tl.col = "black", tl.srt = 45)
```




```
# removing variables that appears collinear with other variables
df <- df |> filter(BMI < 60) |> dplyr::select(-c(MentHlth, PhysHlth, DiffWalk, Education, Income, HighBl
```

Splitting the data

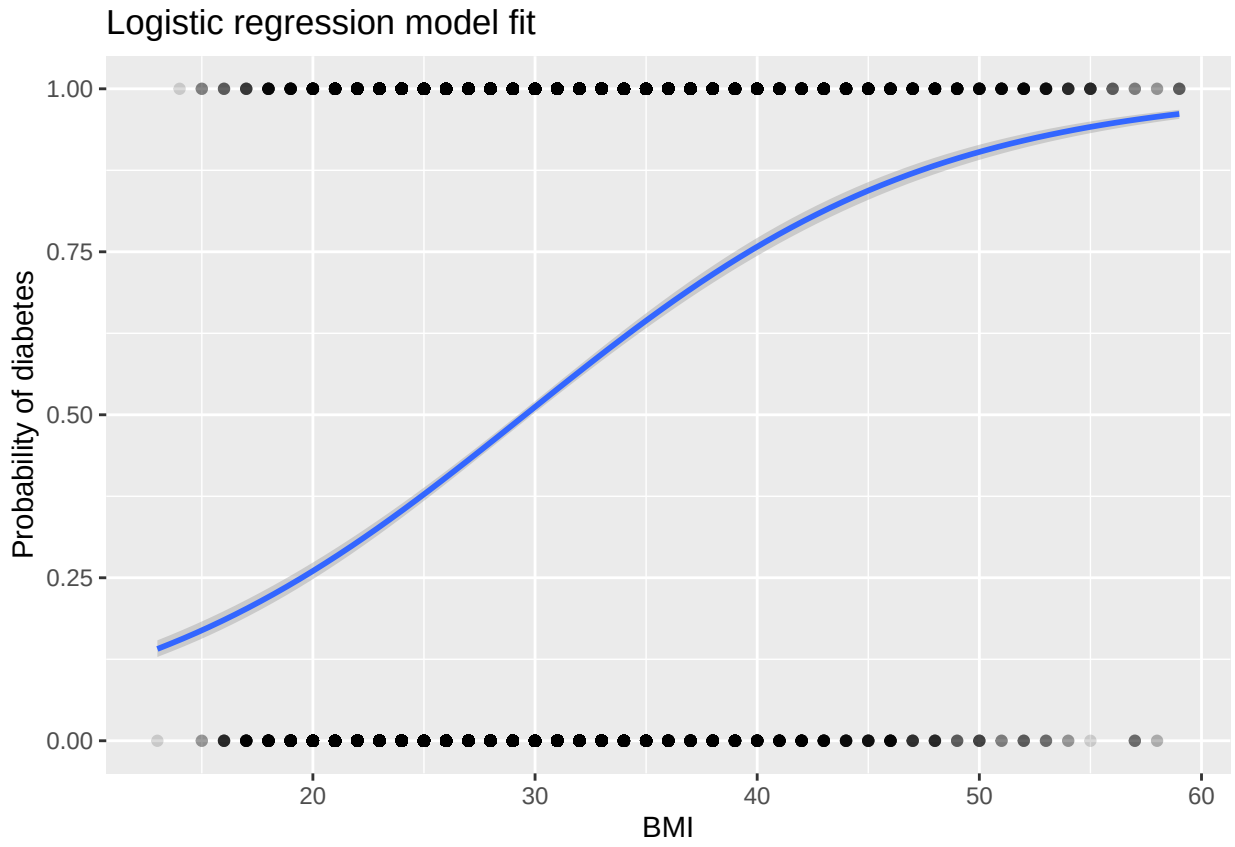
```
y <- df$Diabetes_binary
test_index <- createDataPartition(y, times = 1, p = 0.2, list = FALSE)

test_set <- df[test_index,]
train_set <- df[-test_index,]
```

```
fit_glm <- glm(Diabetes_binary ~ ., data = train_set, family = "binomial")
p_hat_glm <- predict(fit_glm, test_set, type="response")
y_hat_glm <- factor(ifelse(p_hat_glm > 0.5, 1, 0))
confusionMatrix(y_hat_glm, as.factor(test_set$Diabetes_binary))$overall["Accuracy"]
```

```
##  Accuracy
## 0.7414246
```

```
ggplot(aes(x = BMI, y = Diabetes_binary), data = test_set) +
  geom_point(alpha = .15) +
  geom_smooth(method = "glm", method.args = list(family = "binomial")) +
  ggtitle("Logistic regression model fit") +
  xlab("BMI") +
  ylab("Probability of diabetes")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

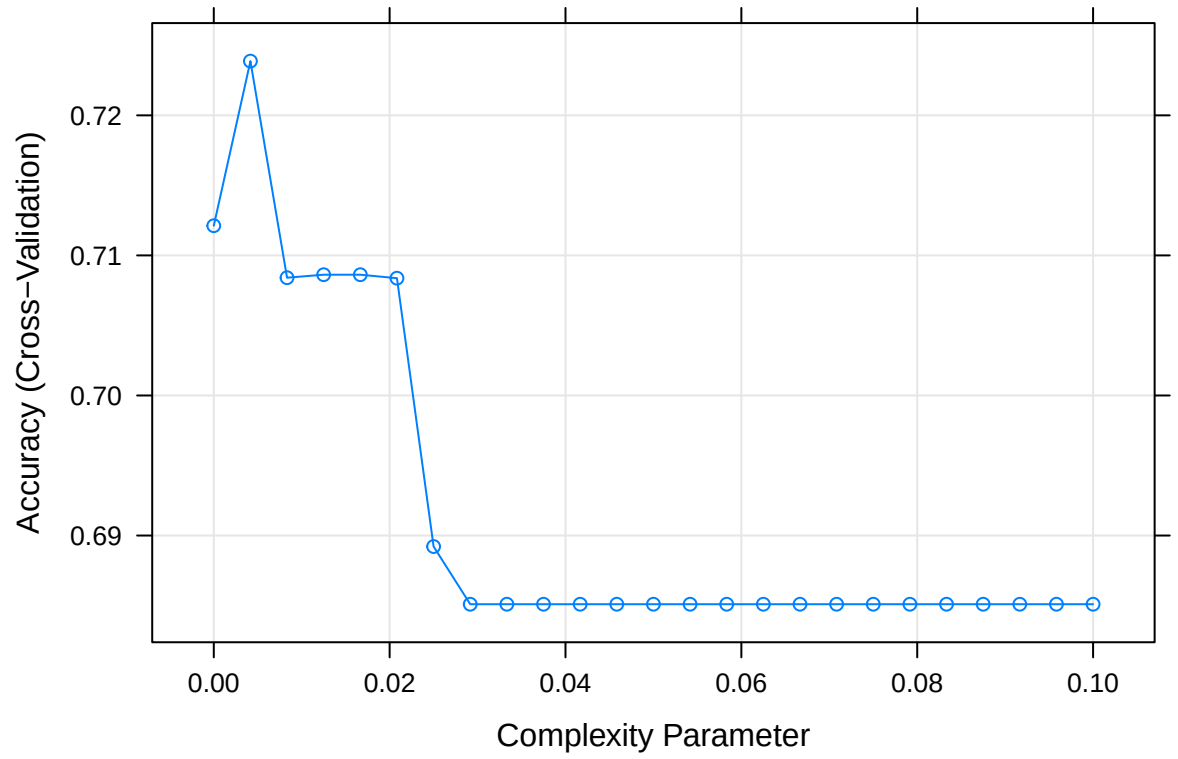



```
summary(fit_glm)
```

```
##
## Call:
## glm(formula = Diabetes_binary ~ ., family = "binomial", data = train_set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1186  -0.8419  -0.1119   0.8709   2.9697
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -8.173637   0.130003 -62.873  < 2e-16 ***
## HighChol                0.680323   0.020584  33.052  < 2e-16 ***
## CholCheck               1.507764   0.092284  16.338  < 2e-16 ***
## BMI                     0.096741   0.001798  53.810  < 2e-16 ***
## Smoker                  0.005901   0.020716   0.285   0.7758
## Stroke                  0.262609   0.045513   5.770 7.93e-09 ***
## HeartDiseaseorAttack    0.315761   0.031539  10.012  < 2e-16 ***
## PhysActivity           -0.049203   0.023203  -2.121   0.0340 *
## Fruits                 -0.046657   0.021666  -2.154   0.0313 *
## Veggies                -0.131081   0.025618  -5.117 3.11e-07 ***
## HvyAlcoholConsump      -0.734303   0.054019 -13.593  < 2e-16 ***
```

```
## AnyHealthcare        -0.026719   0.051861  -0.515   0.6064
## NoDocbcCost           0.062184   0.037254   1.669   0.0951 .
## GenHlth               0.628827   0.010902  57.678  < 2e-16 ***
## Sex                   0.244472   0.020672  11.826  < 2e-16 ***
## Age                   0.197412   0.004160  47.459  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 78079  on 56321  degrees of freedom
## Residual deviance: 58776  on 56306  degrees of freedom
## AIC: 58808
##
## Number of Fisher Scoring iterations: 5
```

```
control <- trainControl(method = "cv", number = 10, p = .9)
train_rpart <- train(as.factor(Diabetes_binary) ~ .,
                     method = "rpart",
                     tuneGrid = data.frame(cp = seq(0.0, 0.1, len = 25)),
                     data = train_set, trControl = control)
plot(train_rpart)
```




```
y_hat <- predict(train_rpart, test_set)
confusionMatrix(y_hat, as.factor(test_set$Diabetes_binary))$overall["Accuracy"]
```

```
##  Accuracy
## 0.7295647
```

```
plot(train_rpart$finalModel, margin = 0.01)
text(train_rpart$finalModel, cex = 0.75)
```