Diamonds Prices

Chu,Li,Hsu

2024-12-28

Table of contents

```
library(showtext)
```

```
Loading required package: sysfonts
Loading required package: showtextdb
```

```r
showtext_auto()  # 啟用 showtext
font_add("Microsoft JhengHei UI", "C:/Windows/Fonts/msjh.ttc")  # 添加你使用的字體
```

```r
library(Hmisc)
```

```

Attaching package: 'Hmisc'

The following objects are masked from 'package:base':

    format.pval, units
```

```r
library(skimr)
library(DataExplorer)
library(ggplot2)
library(dplyr)
```

```

Attaching package: 'dplyr'

The following objects are masked from 'package:Hmisc':

    src, summarize

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```r
library(corrplot)
```

```
corrplot 0.92 loaded
```

```r
library(GGally)
```

```
Registered S3 method overwritten by 'GGally':
  method from
  +.gg   ggplot2
```

```r
library(plotly)
```

```

Attaching package: 'plotly'

The following object is masked from 'package:ggplot2':

    last_plot

The following object is masked from 'package:Hmisc':

    subplot

The following object is masked from 'package:stats':
    filter

The following object is masked from 'package:graphics':

    layout
```

```r
library(gridExtra)
```

```

Attaching package: 'gridExtra'

The following object is masked from 'package:dplyr':

    combine
```

```r
library(knitr)
library(car)
```

```
Loading required package: carData


Attaching package: 'car'

The following object is masked from 'package:dplyr':

    recode
```

```r
setwd("C:/Users/User/OneDrive/桌面/統諮期末")
data <- read.csv("Sleep_health_and_lifestyle_dataset.csv")
```

1. Conduct necessary data preprocessing

敘述性統計/missing values 診斷

```r
# Check structure of the dataset
head(data)
```

```
  Person.ID Gender Age           Occupation Sleep.Duration Quality.of.Sleep
1         1   Male  27    Software Engineer            6.1                6
2         2   Male  28               Doctor            6.2                6
3         3   Male  28               Doctor            6.2                6
4         4   Male  28 Sales Representative            5.9                4
5         5   Male  28 Sales Representative            5.9                4
6         6   Male  28    Software Engineer            5.9                4
  Physical.Activity.Level Stress.Level BMI.Category Blood.Pressure Heart.Rate
1                      42            6   Overweight         126/83         77
2                      60            8       Normal         125/80         75
3                      60            8       Normal         125/80         75
4                      30            8        Obese         140/90         85
5                      30            8        Obese         140/90         85
6                      30            8        Obese         140/90         85
  Daily.Steps Sleep.Disorder
```

```
1        4200           None
2       10000           None
3       10000           None
4        3000    Sleep Apnea
5        3000    Sleep Apnea
6        3000       Insomnia
```

```r
dim(data)
```

```
[1] 374  13
```

```r
names(data)
```

```
 [1] "Person.ID"               "Gender"
 [3] "Age"                     "Occupation"
 [5] "Sleep.Duration"          "Quality.of.Sleep"
 [7] "Physical.Activity.Level" "Stress.Level"
 [9] "BMI.Category"            "Blood.Pressure"
[11] "Heart.Rate"              "Daily.Steps"
[13] "Sleep.Disorder"
```

```r
str(data)
```

```
'data.frame':   374 obs. of  13 variables:
 $ Person.ID              : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Gender                 : chr  "Male" "Male" "Male" "Male" ...
 $ Age                    : int  27 28 28 28 28 28 29 29 29 29 ...
 $ Occupation             : chr  "Software Engineer" "Doctor" "Doctor" "Sales Representa
 $ Sleep.Duration         : num  6.1 6.2 6.2 5.9 5.9 5.9 6.3 7.8 7.8 7.8 ...
 $ Quality.of.Sleep       : int  6 6 6 4 4 4 6 7 7 7 ...
 $ Physical.Activity.Level: int  42 60 60 30 30 30 40 75 75 75 ...
 $ Stress.Level           : int  6 8 8 8 8 8 7 6 6 6 ...
 $ BMI.Category           : chr  "Overweight" "Normal" "Normal" "Obese" ...
 $ Blood.Pressure         : chr  "126/83" "125/80" "125/80" "140/90" ...
 $ Heart.Rate             : int  77 75 75 85 85 85 82 70 70 70 ...
 $ Daily.Steps            : int  4200 10000 10000 3000 3000 3000 3500 8000 8000 8000 ...
 $ Sleep.Disorder         : chr  "None" "None" "None" "Sleep Apnea" ...
```

```r
skim(data)
```

Table 1: Data summary

Name	data
Number of rows	374
Number of columns	13
Column type frequency:	
character	5
numeric	8
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Gender	0	1	4	6	0	2	0
Occupation	0	1	5	20	0	11	0
BMI.Category	0	1	5	13	0	4	0
Blood.Pressure	0	1	6	6	0	25	0
Sleep.Disorder	0	1	4	11	0	3	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Person.ID	0	1	187.50	108.11	1.0	94.25	187.5	280.75	374.0	
Age	0	1	42.18	8.67	27.0	35.25	43.0	50.00	59.0	
Sleep.Duration	0	1	7.13	0.80	5.8	6.40	7.2	7.80	8.5	
Quality.of.Sleep	0	1	7.31	1.20	4.0	6.00	7.0	8.00	9.0	
Physical.Activity.Level	0	1	59.17	20.83	30.0	45.00	60.0	75.00	90.0	
Stress.Level	0	1	5.39	1.77	3.0	4.00	5.0	7.00	8.0	
Heart.Rate	0	1	70.17	4.14	65.0	68.00	70.0	72.00	86.0	
Daily.Steps	0	1	6816.84	1617.92	3000.0	5600.00	7000.0	8000.00	10000.0	

```r
describe(data)
```

```
data

 13  Variables      374  Observations
--------------------------------------------------------------------------------
Person.ID
       n  missing distinct     Info     Mean      Gmd      .05      .10
     374        0      374        1    187.5      125    19.65    38.30
     .25      .50      .75      .90      .95
   94.25   187.50   280.75   336.70   355.35

lowest :   1   2   3   4   5, highest: 370 371 372 373 374
--------------------------------------------------------------------------------
Gender
       n  missing distinct
     374        0        2

Value      Female   Male
Frequency     185    189
Proportion  0.495  0.505
--------------------------------------------------------------------------------
Age
       n  missing distinct     Info     Mean      Gmd      .05      .10
     374        0       31    0.997    42.18    9.933    29.65    31.00
     .25      .50      .75      .90      .95
   35.25    43.00    50.00    54.00    58.00

lowest : 27 28 29 30 31, highest: 55 56 57 58 59
```

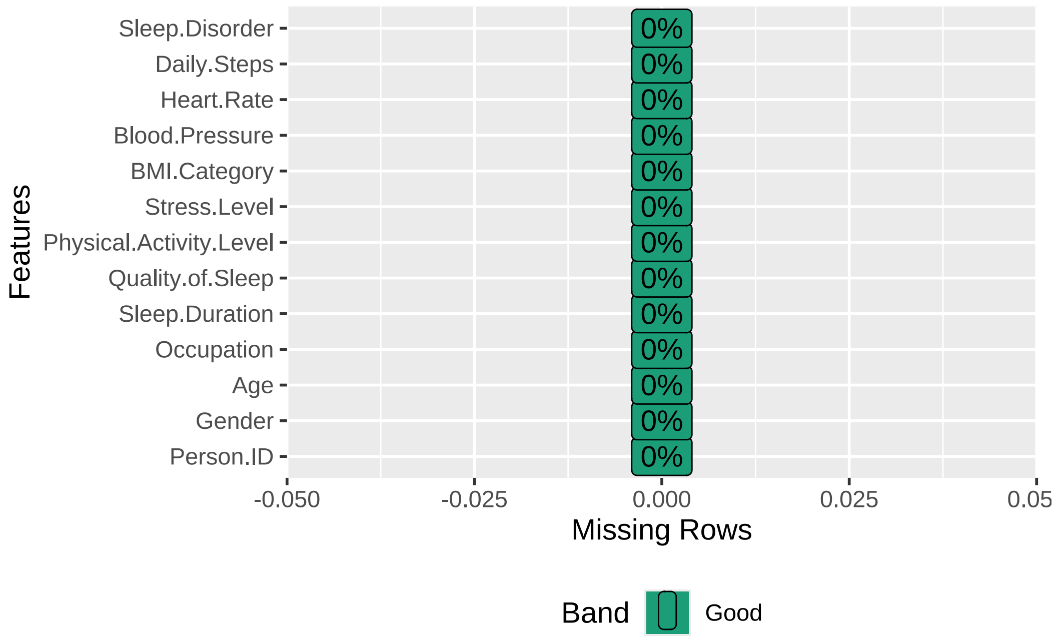
```
--------------------------------------------------------------------------------
Occupation
       n  missing distinct
     374        0       11

lowest : Accountant           Doctor               Engineer             Lawyer
highest: Sales Representative Salesperson          Scientist            Software Enginee
--------------------------------------------------------------------------------
Sleep.Duration
       n  missing distinct     Info     Mean      Gmd      .05      .10
     374        0       27    0.997    7.132   0.9153      6.0      6.1
     .25      .50      .75      .90      .95
     6.4      7.2      7.8      8.2      8.4

lowest : 5.8 5.9 6   6.1 6.2, highest: 8.1 8.2 8.3 8.4 8.5
--------------------------------------------------------------------------------
Quality.of.Sleep
       n  missing distinct     Info     Mean      Gmd
     374        0        6    0.938    7.313    1.329

Value          4     5     6     7     8     9
Frequency      5     7   105    77   109    71
Proportion 0.013 0.019 0.281 0.206 0.291 0.190

For the frequency table, variable is rounded to the nearest 0
--------------------------------------------------------------------------------
Physical.Activity.Level
       n  missing distinct     Info     Mean      Gmd      .05      .10
     374        0       16     0.97    59.17    23.69       30       30
     .25      .50      .75      .90      .95
      45       60       75       90       90

Value         30    32    35    40    42    45    47    50    55    60    65
Frequency     68     2     4     6     2    68     1     4     6    70     2
Proportion 0.182 0.005 0.011 0.016 0.005 0.182 0.003 0.011 0.016 0.187 0.005

Value         70    75    80    85    90
Frequency      3    67     2     2    67
Proportion 0.008 0.179 0.005 0.005 0.179

For the frequency table, variable is rounded to the nearest 0
--------------------------------------------------------------------------------
Stress.Level
       n  missing distinct     Info     Mean      Gmd
     374        0        6     0.97    5.385    2.017

Value          3     4     5     6     7     8
Frequency     71    70    67    46    50    70
Proportion 0.190 0.187 0.179 0.123 0.134 0.187
```

```
For the frequency table, variable is rounded to the nearest 0
--------------------------------------------------------------------------------
BMI.Category
       n  missing distinct 
     374        0        4 
                                                              
Value              Normal Normal Weight         Obese    Overweight
Frequency             195            21            10           148
Proportion          0.521         0.056         0.027         0.396
--------------------------------------------------------------------------------
Blood.Pressure
       n  missing distinct 
     374        0       25 

lowest : 115/75 115/78 117/76 118/75 118/76, highest: 135/90 139/91 140/90 140/95 142/92
--------------------------------------------------------------------------------
Heart.Rate
       n  missing distinct     Info     Mean      Gmd      .05      .10 
     374        0       19    0.963    70.17    4.353       65       65 
     .25      .50      .75      .90      .95 
      68       70       72       75       78 
                                                                            
Value         65    67    68    69    70    72    73    74    75    76    77
Frequency     67     2    94     2    76    69     2     2    36     2     2
Proportion 0.179 0.005 0.251 0.005 0.203 0.184 0.005 0.005 0.096 0.005 0.005
                                                  
Value         78    80    81    82    83    84    85    86
Frequency      5     3     2     1     2     2     3     2
Proportion 0.013 0.008 0.005 0.003 0.005 0.005 0.008 0.005

For the frequency table, variable is rounded to the nearest 0
--------------------------------------------------------------------------------
Daily.Steps
       n  missing distinct     Info     Mean      Gmd      .05      .10 
     374        0       20    0.962     6817     1801     4930     5000 
     .25      .50      .75      .90      .95 
    5600     7000     8000     8000    10000 
                                                                            
Value       3000  3300  3500  3700  4000  4100  4200  4800  5000  5200  5500
Frequency      3     2     3     2     3     2     2     2    68     2     4
Proportion 0.008 0.005 0.008 0.005 0.008 0.005 0.005 0.005 0.182 0.005 0.011
                                                              
Value       5600  6000  6200  6800  7000  7300  7500  8000 10000
Frequency      2    68     1     3    66     2     2   101    36
Proportion 0.005 0.182 0.003 0.008 0.176 0.005 0.005 0.270 0.096

For the frequency table, variable is rounded to the nearest 0
--------------------------------------------------------------------------------
Sleep.Disorder
       n  missing distinct 
     374        0        3

Value          Insomnia         None  Sleep  Apnea
Frequency            77          219            78
Proportion        0.206        0.586         0.209
--------------------------------------------------------------------------------
```

```r
sum(is.na(data))
```

```
[1] 0
```

```r
plot_missing(data)
```



此筆資料集共有 374 筆資料，13 個變數且無缺失值

其中 gender,occupation,quality.of.sleep 為類別變數;

age,sleep.duration,blood.pressure 為連續變數

Scaling for predicting

```r
# Scale numerical variables
#num_cols <- c("carat", "depth", "table", "price", "x", "y", "z")
#diamond[num_cols] <- scale(diamond[num_cols])
```

table one

```r
summary_table <- data %>%
  summarise(
    Variable = c(
      "Person ID",
      "Gender",
      "Age",
      "Occupation",
      "Sleep Duration",
      "Quality of Sleep",
      "Physical Activity Level",
      "Stress Level",
      "BMI Category",
      "Blood Pressure",
      "Heart Rate",
      "Daily Steps",
      "Sleep Disorder"
    ),
    Description = c(
      "編號",
      "性別",
      "年齡",
      "職業",
      "每日睡眠時長(小時)",
      "主觀認定之睡眠品質",
      "Physical Activity Level",
      "主觀認定之壓力程度",
      "BMI類別",
      "血壓",
      "脈搏",
      "每日步數",
      "睡眠疾病"
    ),
    remark=c(
      "1-374",
      "Male/Female",
      "27-59",
      "Occupation",
      "Sleep Duration",
      "scale: 1-10",
      "Physical Activity Level",
      "scale: 1-10",
      "Underweight/Normal/Overweight...",
      "systolic/diastolic",
      "bpm",
      "Daily Steps",
      "None/Insomnia/Apnea"
    )
  )
```

```
Warning: Returning more (or less) than 1 row per `summarise()` group was deprecated in
dplyr 1.1.0.
i Please use `reframe()` instead.
i When switching from `summarise()` to `reframe()`, remember that `reframe()`
  always returns an ungrouped data frame and adjust accordingly.
```

```
kable(summary_table, format = "markdown", digits = 2, caption = "變數解釋")
```

Table 4: 變數解釋

Variable	Description	remark
Person ID	編號	1-374
Gender	性別	Male/Female
Age	年齡	27-59
Occupation	職業	Occupation
Sleep Duration	每日睡眠時長 (小時)	Sleep Duration
Quality of Sleep	主觀認定之睡眠品質	scale: 1-10
Physical Activity Level	Physical Activity Level	Physical Activity Level
Stress Level	主觀認定之壓力程度	scale: 1-10
BMI Category	BMI 類別	Underweight/Normal/Overweight...
Blood Pressure	血壓	systolic/diastolic
Heart Rate	脈搏	bpm
Daily Steps	每日步數	Daily Steps
Sleep Disorder	睡眠疾病	None/Insomnia/Apnea

資料前處理

```
# 刪除 Person ID
data <- data %>% select(-`Person.ID`)

# 把 blood pressure 分成兩 col
data <- data %>%
  tidyr::separate(col = `Blood.Pressure`,
                  into = c("BloodPressure_Upper", "BloodPressure_Lower"),
                  sep = "/",
                  convert = TRUE) # convert=TRUE 會自動轉換為數值型別
data$BloodPressure_Upper <- as.numeric(data$BloodPressure_Upper)
data$BloodPressure_Lower <- as.numeric(data$BloodPressure_Lower)

# 分類 physical activity level
data$Physical.Activity.Level<-ifelse(data$Physical.Activity.Level<=35,"<=35",
                              ifelse(data$Physical.Activity.Level<=45,"<=45",
                              ifelse(data$Physical.Activity.Level<=60,"<=60",
                              ifelse(data$Physical.Activity.Level<=75,"<=75",
                              "<=90"))))
# 分類 daily steps
data$Daily.Steps <- ifelse(data$Daily.Steps<=5000,"<=5000",
                    ifelse(data$Daily.Steps<=6000,"<=6000",
                    ifelse(data$Daily.Steps<=7000,"<=7000","7000up")))

# 將睡眠疾病->0,1
data$Sleep.Disorder <- ifelse(data$Sleep.Disorder=="None",0,1)

# 分類 BMI
data$BMI.Category <- ifelse(data$BMI.Category == "Normal Weight","Normal",
                            data$BMI.Category)
data$BMI.Category <- ifelse(data$BMI.Category == "Obese","Overweight",
                            data$BMI.Category)

# 分類 quality of sleep
data$Quality.of.Sleep <- ifelse(data$Quality.of.Sleep==4 |
                                data$Quality.of.Sleep==5,"4-5",
                                data$Quality.of.Sleep)

# 分類 occupation
data$Occupation <- ifelse(data$Occupation=="Manager" | data$Occupation=="Sales Represent
data$Occupation <- ifelse(data$Occupation=="Software Engineer"  ,"Engineer",data$Occupat
```

Encoding Categorical Variables

```
data$Gender <- as.factor(data$Gender)
data$Occupation <- as.factor(data$Occupation)
data$Quality.of.Sleep <- as.factor(data$Quality.of.Sleep)
data$Stress.Level <- as.factor(data$Stress.Level)
data$BMI.Category <- as.factor(data$BMI.Category)
data$Sleep.Disorder <- as.factor(data$Sleep.Disorder)
data$Physical.Activity.Level <- as.factor(data$Physical.Activity.Level)
data$Daily.Steps <- as.factor(data$Daily.Steps)
```

處理後的資料

```
describe(data)
```

```
data

 13  Variables      374  Observations
--------------------------------------------------------------------------------
Gender
       n  missing distinct 
     374        0        2 

Value      Female   Male
Frequency     185    189
Proportion  0.495  0.505
--------------------------------------------------------------------------------
Age
       n  missing distinct     Info     Mean      Gmd      .05      .10 
     374        0       31    0.997    42.18    9.933    29.65    31.00 
     .25      .50      .75      .90      .95 
   35.25    43.00    50.00    54.00    58.00 

lowest : 27 28 29 30 31, highest: 55 56 57 58 59
--------------------------------------------------------------------------------
Occupation
       n  missing distinct 
     374        0        8 
```

```
Value       Accountant      Doctor    Engineer      Lawyer       Nurse
Frequency           37          71          67          47          73
Proportion       0.099       0.190       0.179       0.126       0.195

Value      Salesperson   Scientist     Teacher
Frequency           35           4          40
Proportion       0.094       0.011       0.107
--------------------------------------------------------------------------------
Sleep.Duration
       n  missing distinct     Info     Mean      Gmd      .05      .10
     374        0       27    0.997    7.132   0.9153      6.0      6.1
     .25      .50      .75      .90      .95
     6.4      7.2      7.8      8.2      8.4

lowest : 5.8 5.9 6   6.1 6.2, highest: 8.1 8.2 8.3 8.4 8.5
--------------------------------------------------------------------------------
Quality.of.Sleep
       n  missing distinct
     374        0        5

Value        4-5     6     7     8     9
Frequency     12   105    77   109    71
Proportion 0.032 0.281 0.206 0.291 0.190
--------------------------------------------------------------------------------
Physical.Activity.Level
       n  missing distinct
     374        0        5

Value       <=35  <=45  <=60  <=75  <=90
Frequency     74    76    81    72    71
Proportion 0.198 0.203 0.217 0.193 0.190
--------------------------------------------------------------------------------
Stress.Level
       n  missing distinct
     374        0        6

Value          3     4     5     6     7     8
Frequency     71    70    67    46    50    70
Proportion 0.190 0.187 0.179 0.123 0.134 0.187
--------------------------------------------------------------------------------
BMI.Category
       n  missing distinct
     374        0        2

Value          Normal Overweight
Frequency         216        158
Proportion      0.578      0.422
--------------------------------------------------------------------------------
BloodPressure_Upper
```

```
       n  missing distinct     Info     Mean      Gmd      .05      .10 
     374        0       18    0.965    128.6     8.74      115      118 
     .25      .50      .75      .90      .95 
     125      130      135      140      140 

Value        115   117   118   119   120   121   122   125   126   128   129
Frequency     34     2     3     2    45     1     1    69     2     5     2
Proportion 0.091 0.005 0.008 0.005 0.120 0.003 0.003 0.184 0.005 0.013 0.005

Value        130   131   132   135   139   140   142
Frequency    101     2     3    29     2    69     2
Proportion 0.270 0.005 0.008 0.078 0.005 0.184 0.005

For the frequency table, variable is rounded to the nearest 0
--------------------------------------------------------------------------------
BloodPressure_Lower 
       n  missing distinct     Info     Mean      Gmd      .05      .10 
     374        0       17    0.947    84.65    6.832       75       77 
     .25      .50      .75      .90      .95 
      80       85       90       95       95 

Value         75    76    77    78    79    80    82    83    84    85    86
Frequency     34     3     2     2     1   111     4     2     4   102     4
Proportion 0.091 0.008 0.005 0.005 0.003 0.297 0.011 0.005 0.011 0.273 0.011

Value         87    88    90    91    92    95
Frequency      3     2    31     2     2    65
Proportion 0.008 0.005 0.083 0.005 0.005 0.174

For the frequency table, variable is rounded to the nearest 0
--------------------------------------------------------------------------------
Heart.Rate 
       n  missing distinct     Info     Mean      Gmd      .05      .10 
     374        0       19    0.963    70.17    4.353       65       65 
     .25      .50      .75      .90      .95 
      68       70       72       75       78 

Value         65    67    68    69    70    72    73    74    75    76    77
Frequency     67     2    94     2    76    69     2     2    36     2     2
Proportion 0.179 0.005 0.251 0.005 0.203 0.184 0.005 0.005 0.096 0.005 0.005

Value         78    80    81    82    83    84    85    86
Frequency      5     3     2     1     2     2     3     2
Proportion 0.013 0.008 0.005 0.003 0.005 0.005 0.008 0.005

For the frequency table, variable is rounded to the nearest 0
--------------------------------------------------------------------------------
Daily.Steps 
       n  missing distinct 
     374        0        4 
```

```
Value      <=5000 <=6000 <=7000 7000up
Frequency      87     76     70    141
Proportion  0.233  0.203  0.187  0.377
--------------------------------------------------------------------------------
Sleep.Disorder
       n  missing distinct
     374        0        2

Value          0     1
Frequency    219   155
Proportion 0.586 0.414
--------------------------------------------------------------------------------
```

描述性統計: 比較不同組別間的變數分布差異

```r
library(Hmisc)
output0 <- summaryM(Age + Gender + Occupation + Sleep.Duration +
             Quality.of.Sleep + Physical.Activity.Level + Stress.Level +
             BMI.Category + BloodPressure_Upper + BloodPressure_Lower +
             Heart.Rate + Daily.Steps
                                 ~ Sleep.Disorder,
                                 #~ 1,
                                 data = data, test = F, overall = F, na.include=T)
#sink(paste0("Table1.txt"))
print(output0, long=TRUE, what = "%")
```

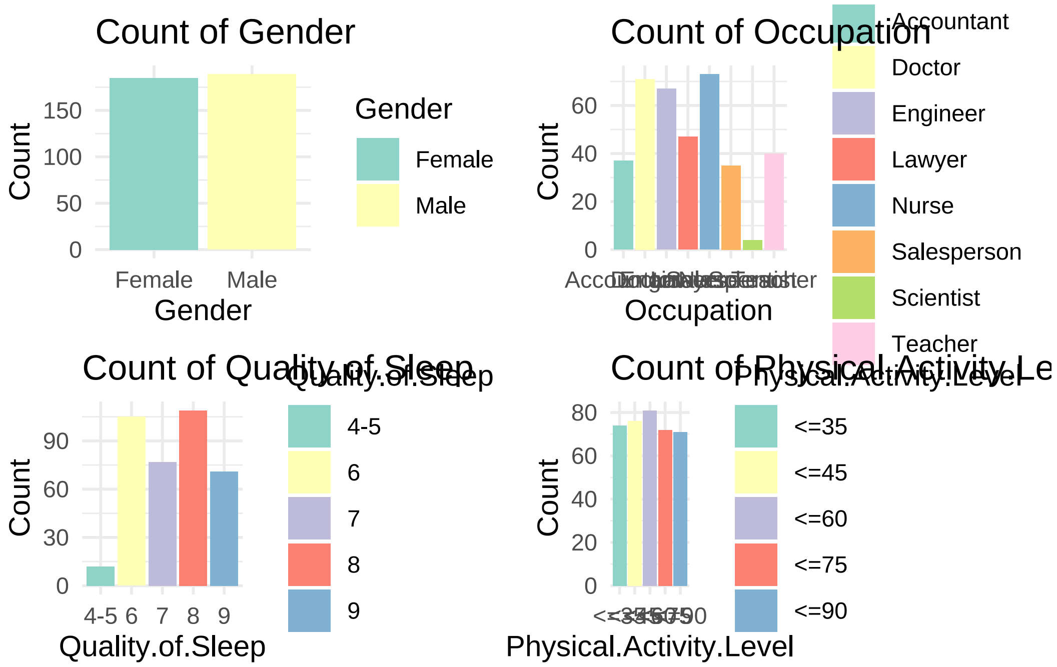
```
Descriptive Statistics  (N=374)

+-----------------------+-----------+-----------+
|                       |0          |1          |
|                       |(N=219)    |(N=155)    |
+-----------------------+-----------+-----------+
|Age                    |  32/38/43 |  43/45/51 |
+-----------------------+-----------+-----------+
|Gender                 |           |           |
+-----------------------+-----------+-----------+
|    Male               | 63%  (137)| 34%  ( 52)|
+-----------------------+-----------+-----------+
|Occupation             |           |           |
+-----------------------+-----------+-----------+
|    Accountant         | 14%  (30) |  5%  ( 7) |
+-----------------------+-----------+-----------+
|    Doctor             | 29%  (64) |  5%  ( 7) |
+-----------------------+-----------+-----------+
|    Engineer           | 27%  (60) |  5%  ( 7) |
+-----------------------+-----------+-----------+
|    Lawyer             | 19%  (42) |  3%  ( 5) |
+-----------------------+-----------+-----------+
|    Nurse              |  4%  ( 9) | 41%  (64) |
+-----------------------+-----------+-----------+
|    Salesperson        |  1%  ( 3) | 21%  (32) |
+-----------------------+-----------+-----------+
|    Scientist          |  1%  ( 2) |  1%  ( 2) |
+-----------------------+-----------+-----------+
|    Teacher            |  4%  ( 9) | 20%  (31) |
+-----------------------+-----------+-----------+
|Sleep.Duration         |7.1/7.4/7.8|6.3/6.5/7.4|
+-----------------------+-----------+-----------+
|Quality.of.Sleep       |           |           |
+-----------------------+-----------+-----------+
|    4-5                |  0%  (  0)|  8%  ( 12)|
+-----------------------+-----------+-----------+
|    6                  | 18%  ( 40)| 42%  ( 65)|
+-----------------------+-----------+-----------+
|    7                  | 18%  ( 40)| 24%  ( 37)|
+-----------------------+-----------+-----------+
|    8                  | 46%  (101)|  5%  (  8)|
+-----------------------+-----------+-----------+
|    9                  | 17%  ( 38)| 21%  ( 33)|
+-----------------------+-----------+-----------+
|Physical.Activity.Level|           |           |
+-----------------------+-----------+-----------+
|    <=35               | 27%  (60) |  9%  (14) |
+-----------------------+-----------+-----------+
|    <=45               |  5%  (10) | 43%  (66) |
+-----------------------+-----------+-----------+
|    <=60               | 34%  (75) |  4%  ( 6) |
+-----------------------+-----------+-----------+
|    <=75               | 18%  (39) | 21%  (33) |
+-----------------------+-----------+-----------+
|    <=90               | 16%  (35) | 23%  (36) |
+-----------------------+-----------+-----------+
|Stress.Level           |           |           |
+-----------------------+-----------+-----------+
|    3                  | 18%  (40) | 20%  (31) |
+-----------------------+-----------+-----------+
|    4                  | 20%  (43) | 17%  (27) |
+-----------------------+-----------+-----------+
|    5                  | 26%  (57) |  6%  (10) |
+-----------------------+-----------+-----------+
|    6                  | 20%  (43) |  2%  ( 3) |
+-----------------------+-----------+-----------+
|    7                  |  1%  ( 3) | 30%  (47) |
+-----------------------+-----------+-----------+
|    8                  | 15%  (33) | 24%  (37) |
+-----------------------+-----------+-----------+
|BMI.Category           |           |           |
+-----------------------+-----------+-----------+
|    Overweight          |  9%  ( 19)| 90%  (139)|
+------------------------+-----------+-----------+
|BloodPressure_Upper     |120/125/130|130/135/140|
+------------------------+-----------+-----------+
|BloodPressure_Lower     |  80/80/85 |  85/90/95 |
+------------------------+-----------+-----------+
|Heart.Rate              |  68/70/70 |  68/72/75 |
+------------------------+-----------+-----------+
|Daily.Steps             |           |           |
+------------------------+-----------+-----------+
|    <=5000              | 29%  ( 63)| 15%  ( 24)|
+------------------------+-----------+-----------+
|    <=6000              |  6%  ( 13)| 41%  ( 63)|
+------------------------+-----------+-----------+
|    <=7000              | 18%  ( 40)| 19%  ( 30)|
+------------------------+-----------+-----------+
|    7000up              | 47%  (103)| 25%  ( 38)|
+------------------------+-----------+-----------+
```

2. EDA

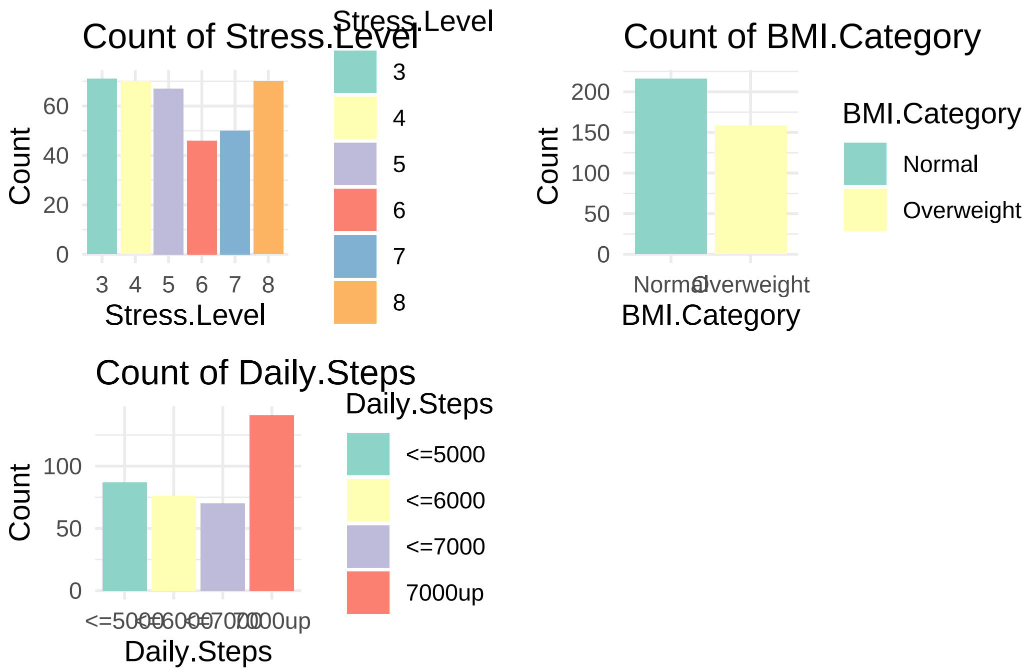
Distribution of the data

i.categorical variable

```r
p1 <- ggplot(data, aes(x = Gender, fill = Gender)) +
  geom_bar() +
  labs(title = "Count of Gender", x = "Gender", y = "Count") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
p2 <- ggplot(data, aes(x = Occupation, fill = Occupation)) +
  geom_bar() +
  labs(title = "Count of Occupation", x = "Occupation", y = "Count") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
p3 <- ggplot(data, aes(x = Quality.of.Sleep, fill = Quality.of.Sleep)) +
  geom_bar() +
  labs(title = "Count of Quality.of.Sleep", x = "Quality.of.Sleep", y = "Count") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
p4 <- ggplot(data,
  aes(x = Physical.Activity.Level, fill = Physical.Activity.Level)) +
  geom_bar() +
  labs(title = "Count of Physical.Activity.Level",
  x = "Physical.Activity.Level", y = "Count") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
grid.arrange(p1,p2,p3,p4,ncol = 2)
```



```r
p5 <- ggplot(data, aes(x = Stress.Level, fill = Stress.Level)) +
  geom_bar() +
  labs(title = "Count of Stress.Level", x = "Stress.Level", y = "Count") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
p6 <- ggplot(data, aes(x = BMI.Category, fill = BMI.Category)) +
  geom_bar() +
  labs(title = "Count of BMI.Category", x = "BMI.Category", y = "Count") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
p7 <- ggplot(data, aes(x = Daily.Steps, fill = Daily.Steps)) +
  geom_bar() +
  labs(title = "Count of Daily.Steps", x = "Daily.Steps", y = "Count") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
grid.arrange(p5,p6,p7,ncol = 2)
```



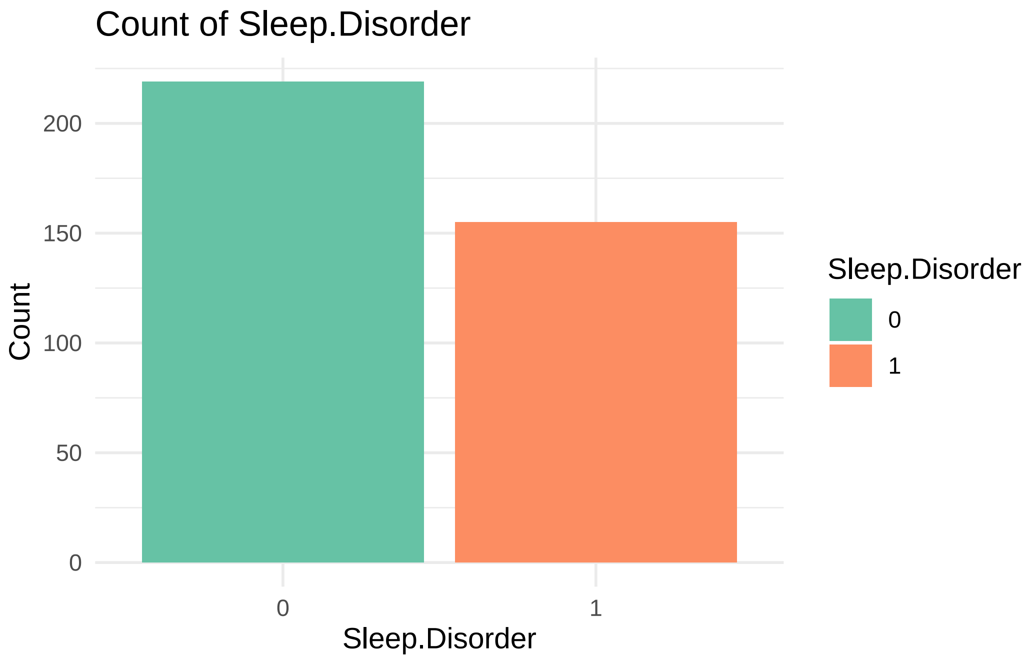
ii.continuous variable

```
layout(mat = matrix(c(1,2),2, byrow = FALSE),  height = c(8,1))
par(mar=c(4, 4, 3, 2))
hist(data$Age, main = 'Distribution of Age',
     xlab="Age",col="lightblue")
par(mar=c(0.5, 4, 0.5, 2))
boxplot(data$Age, xaxt = "n", horizontal=TRUE,
        col="pink", border="black", frame = FALSE)

par(mar=c(4, 4, 3, 2))
hist(data$Sleep.Duration, main = 'Distribution of Sleep.Duration',
     xlab="Sleep.Duration",col="lightblue")
par(mar=c(0.5, 4, 0.5, 2))
boxplot(data$Sleep.Duration, xaxt = "n", horizontal=TRUE,
        col="pink", border="black", frame = FALSE)

par(mar=c(4, 4, 3, 2))
hist(data$BloodPressure_Upper, main = 'Distribution of BloodPressure_Upper',
     xlab="BloodPressure_Upper",col="lightblue")
par(mar=c(0.5, 4, 0.5, 2))
boxplot(data$BloodPressure_Upper, xaxt = "n", horizontal=TRUE,
        col="pink", border="black", frame = FALSE)

par(mar=c(4, 4, 3, 2))
hist(data$BloodPressure_Lower, main = 'Distribution of BloodPressure_Lower',
     xlab="BloodPressure_Lower",col="lightblue")
par(mar=c(0.5, 4, 0.5, 2))
boxplot(data$BloodPressure_Lower, xaxt = "n", horizontal=TRUE,
        col="pink", border="black", frame = FALSE)
```

```
par(mar=c(4, 4, 3, 2))
hist(data$Heart.Rate, main = 'Distribution of Heart.Rate',
     xlab="Heart.Rate",col="lightblue")
par(mar=c(0.5, 4, 0.5, 2))
boxplot(data$Heart.Rate, xaxt = "n", horizontal=TRUE,
        col="pink", border="black", frame = FALSE)
```

iii.Sleep Disorder

```
ggplot(data, aes(x = Sleep.Disorder, fill = Sleep.Disorder)) +
  geom_bar() +
  labs(title = "Count of Sleep.Disorder", x = "Sleep.Disorder", y = "Count") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set2")
```



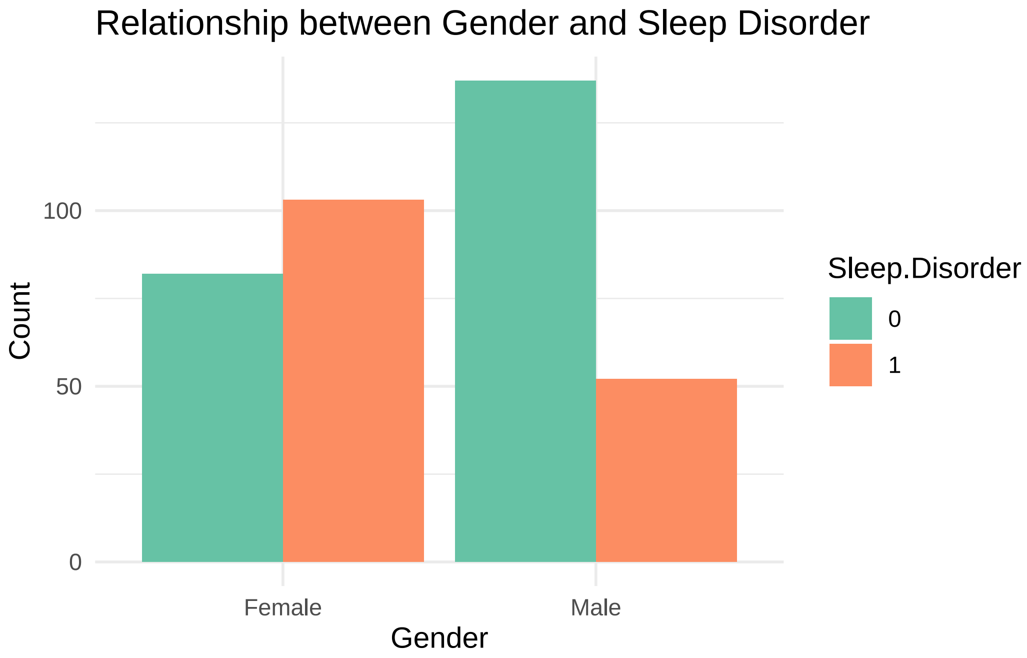
```
data %>%
  group_by(`Sleep.Disorder`, Gender) %>%
  summarise(count = n(), .groups = "drop")
```

```
# A tibble: 4 x 3
  Sleep.Disorder Gender count
  <fct>          <fct>  <int>
1 0              Female    82
2 0              Male     137
3 1              Female   103
4 1              Male      52
```

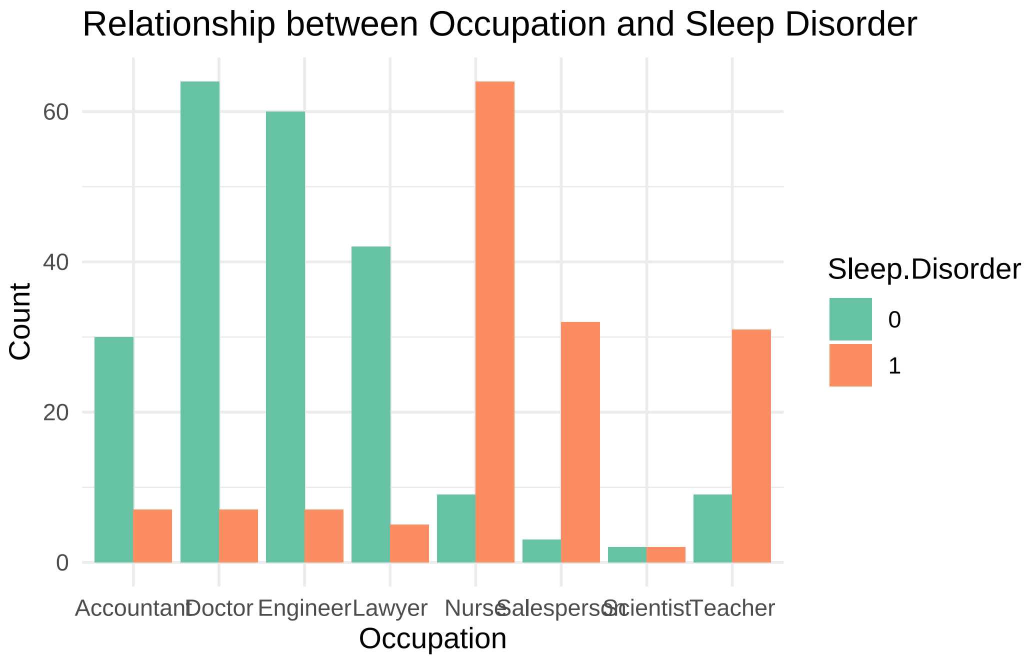
Correlation between data(variables & sleep disorder)

i.categorical variable

```
ggplot(data, aes(x = Gender, fill = Sleep.Disorder)) +
  geom_bar(position = "dodge") +
  labs(title = "Relationship between Gender and Sleep Disorder",
       x = "Gender",
       y = "Count") +
  scale_fill_brewer(palette = "Set2") +
  theme_minimal()
```

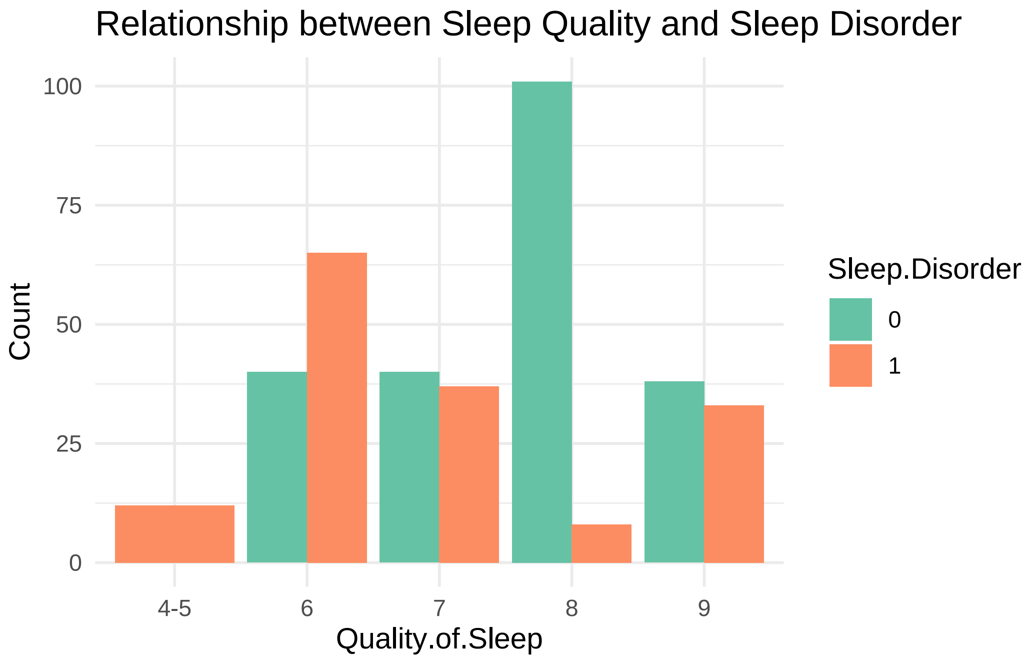


```
ggplot(data, aes(x = Occupation, fill = Sleep.Disorder)) +
  geom_bar(position = "dodge") +
  labs(title = "Relationship between Occupation and Sleep Disorder",
       x = "Occupation",
       y = "Count") +
  scale_fill_brewer(palette = "Set2") +
  theme_minimal()
```



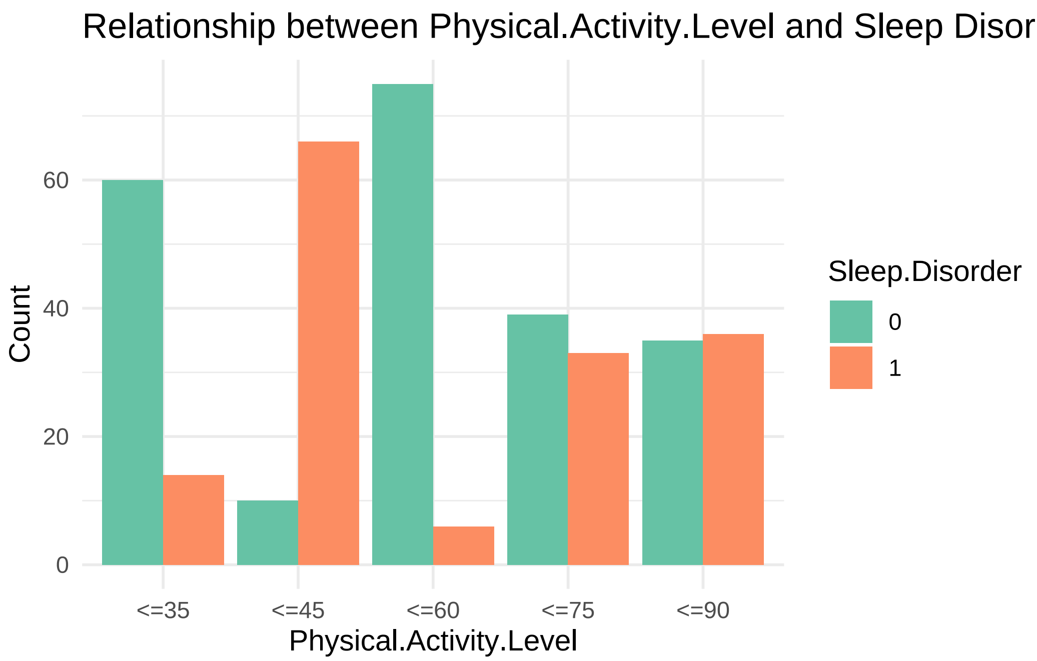
Relationship between Occupation and Sleep Disorder

```
ggplot(data, aes(x = Quality.of.Sleep, fill = Sleep.Disorder)) +
  geom_bar(position = "dodge") +
  labs(title = "Relationship between Sleep Quality and Sleep Disorder",
       x = "Quality.of.Sleep",
       y = "Count") +
  scale_fill_brewer(palette = "Set2") +
  theme_minimal()
```

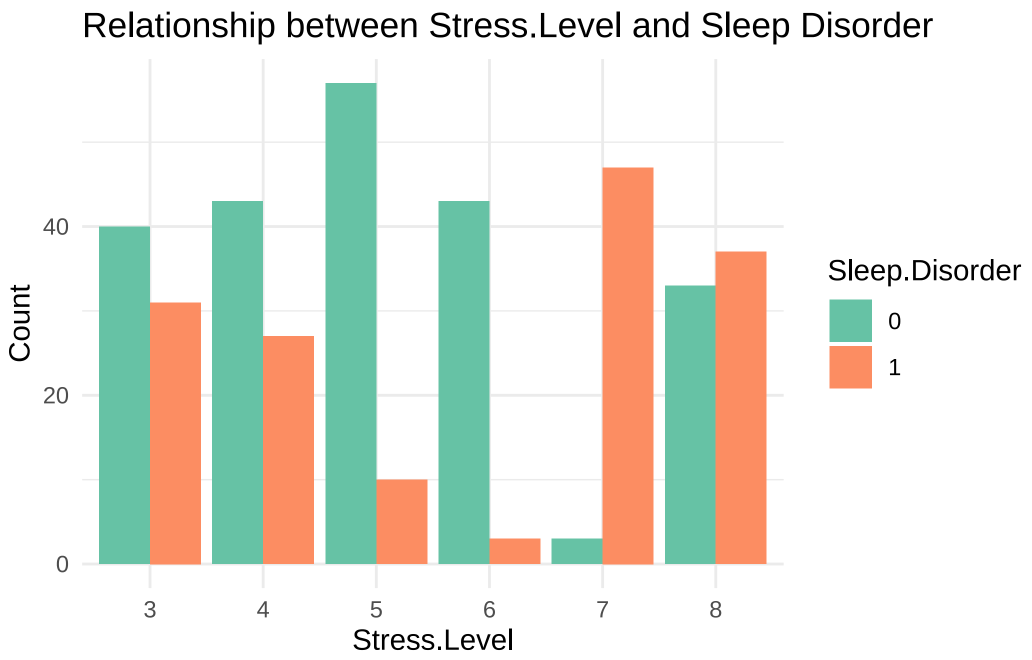


Relationship between Sleep Quality and Sleep Disorder

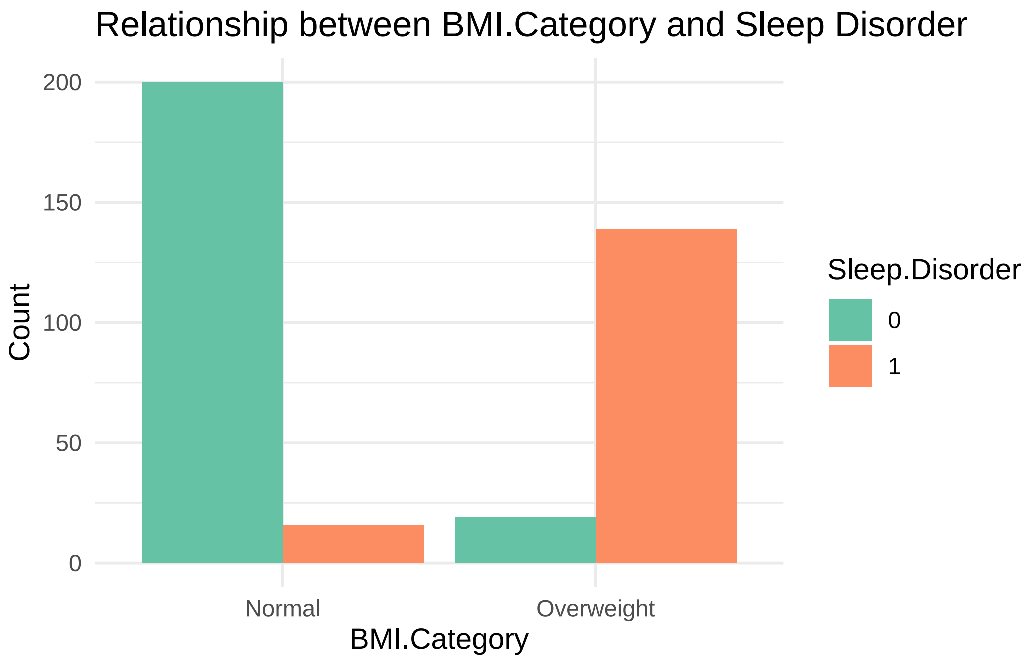
```
ggplot(data, aes(x = Physical.Activity.Level, fill = Sleep.Disorder)) +
  geom_bar(position = "dodge") +
  labs(title = "Relationship between Physical.Activity.Level and Sleep Disorder",
       x = "Physical.Activity.Level",
       y = "Count") +
  scale_fill_brewer(palette = "Set2") +
  theme_minimal()
```



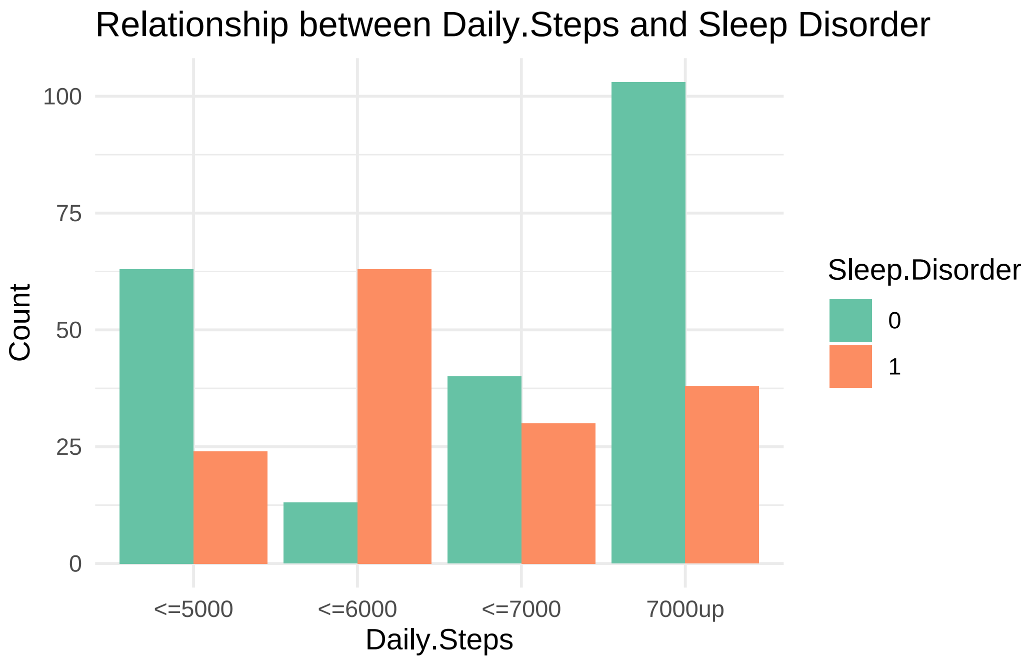
```
ggplot(data, aes(x = Stress.Level, fill = Sleep.Disorder)) +
  geom_bar(position = "dodge") +
  labs(title = "Relationship between Stress.Level and Sleep Disorder",
       x = "Stress.Level",
       y = "Count") +
  scale_fill_brewer(palette = "Set2") +
  theme_minimal()
```



```
ggplot(data, aes(x = BMI.Category, fill = Sleep.Disorder)) +
  geom_bar(position = "dodge") +
  labs(title = "Relationship between BMI.Category and Sleep Disorder",
       x = "BMI.Category",
       y = "Count") +
  scale_fill_brewer(palette = "Set2") +
  theme_minimal()
```



```
ggplot(data, aes(x = Daily.Steps, fill = Sleep.Disorder)) +
  geom_bar(position = "dodge") +
  labs(title = "Relationship between Daily.Steps and Sleep Disorder",
       x = "Daily.Steps",
       y = "Count") +
  scale_fill_brewer(palette = "Set2") +
  theme_minimal()
```



卡方檢定

```
chisq.test(table(data$Occupation, data$Sleep.Disorder))
```

```
Warning in chisq.test(table(data$Occupation, data$Sleep.Disorder)): Chi-squared
approximation may be incorrect


	Pearson's Chi-squared test

data:  table(data$Occupation, data$Sleep.Disorder)
X-squared = 203.69, df = 7, p-value < 2.2e-16
```

```
fisher.test(table(data$Occupation, data$Sleep.Disorder),simulate.p.value=TRUE)
```

```


	Fisher's Exact Test for Count Data with simulated p-value (based on
	2000 replicates)

data:  table(data$Occupation, data$Sleep.Disorder)
p-value = 0.0004998
alternative hypothesis: two.sided
```

```
library(vcd)
```

```
Loading required package: grid
```

```
# 計算 Cramér's V
assocstats(table(data$Gender, data$Sleep.Disorder))$cramer
```

```
[1] 0.2858244
```

偷放幾個酷酷的圖

```
library(ggmosaic)
```

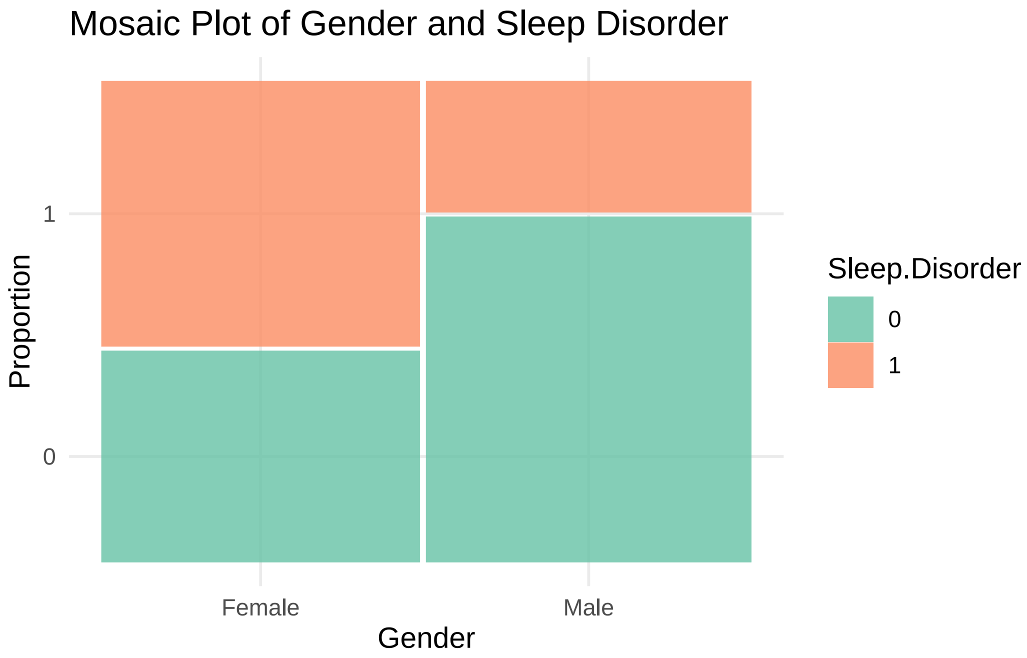
```

Attaching package: 'ggmosaic'

The following objects are masked from 'package:vcd':

    mosaic, spine

The following object is masked from 'package:GGally':

    happy
```

```
# 繪製馬賽克圖
ggplot(data) +
  geom_mosaic(aes(x = product(Gender), fill = Sleep.Disorder)) +
  labs(title = "Mosaic Plot of Gender and Sleep Disorder",
       x = "Gender",
       y = "Proportion") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set2")
```

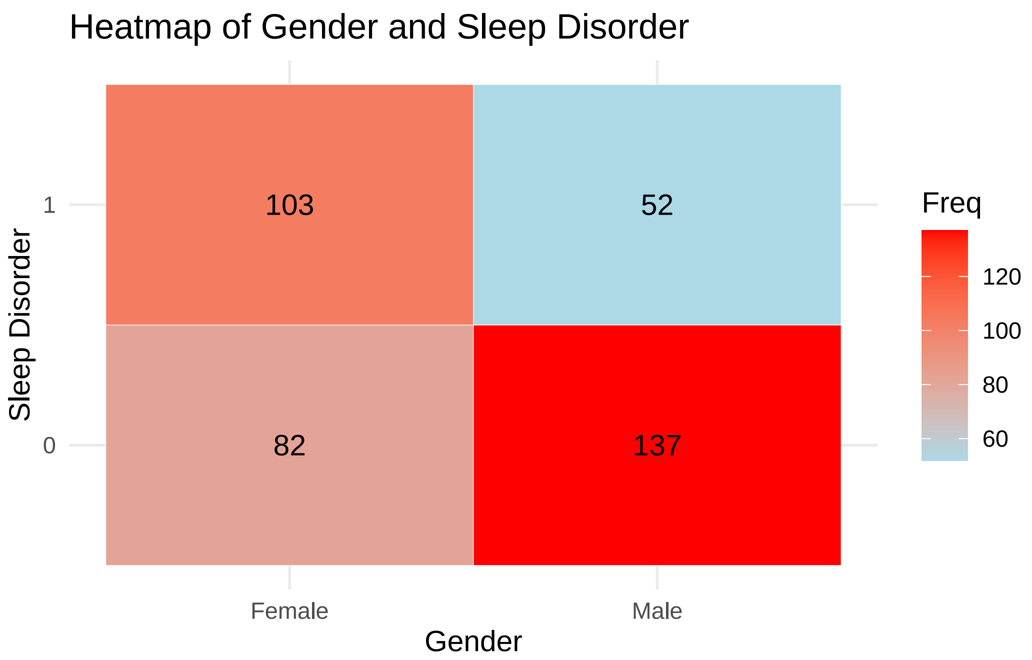
```
Warning: The `scale_name` argument of `continuous_scale()` is deprecated as of ggplot2
3.5.0.
```

```
Warning: The `trans` argument of `continuous_scale()` is deprecated as of ggplot2 3.5.0.
i Please use the `transform` argument instead.
```

```
Warning: `unite_()` was deprecated in tidyr 1.2.0.
i Please use `unite()` instead.
i The deprecated feature was likely used in the ggmosaic package.
  Please report the issue at <https://github.com/haleyjeppson/ggmosaic>.
```

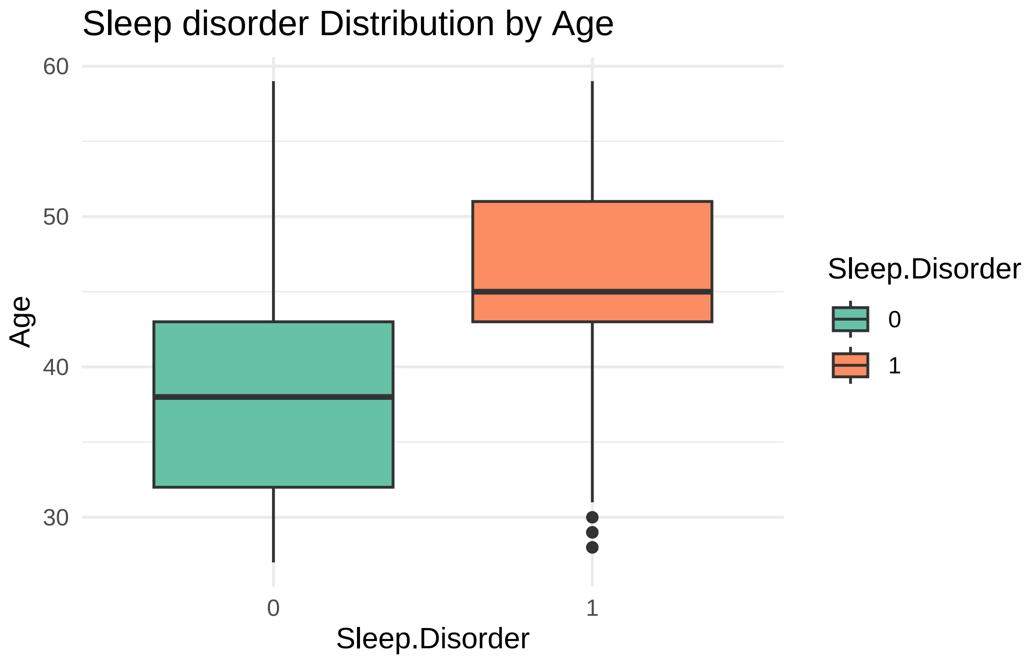


```r
library(reshape2)
# 創建交叉表
table_data <- table(data$Gender, data$Sleep.Disorder)
heatmap_data <- as.data.frame(as.table(table_data))
# 繪製熱圖
ggplot(heatmap_data, aes(x = Var1, y = Var2, fill = Freq)) +
  geom_tile(color = "white") +
  geom_text(aes(label = Freq), color = "black") +
  scale_fill_gradient(low = "lightblue", high = "red") +
  labs(title = "Heatmap of Gender and Sleep Disorder",
       x = "Gender",
       y = "Sleep Disorder") +
  theme_minimal()
```

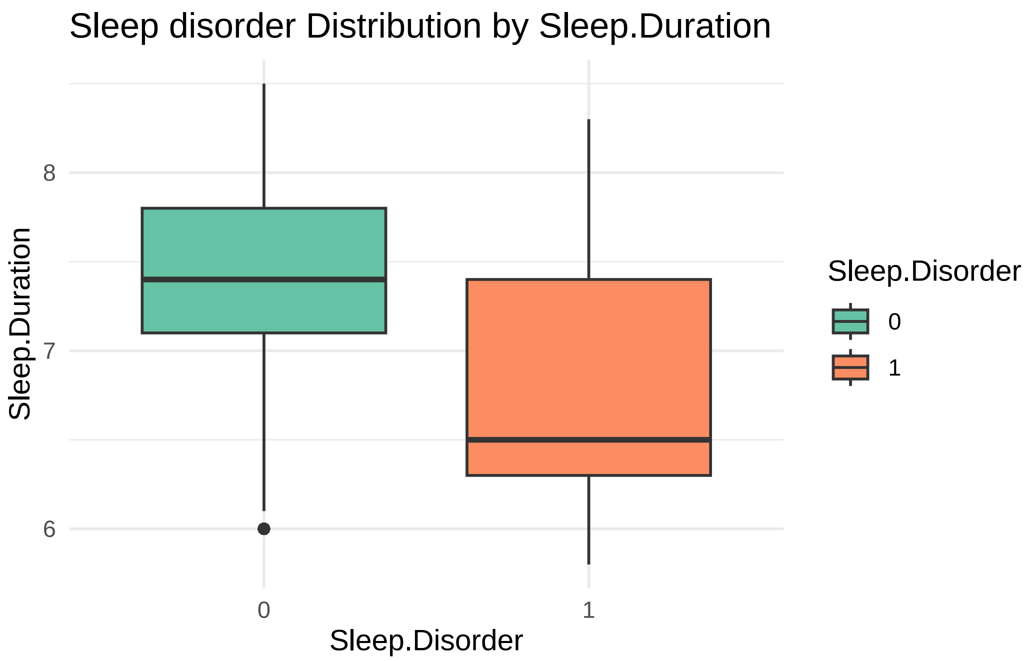


ii.continuous variable

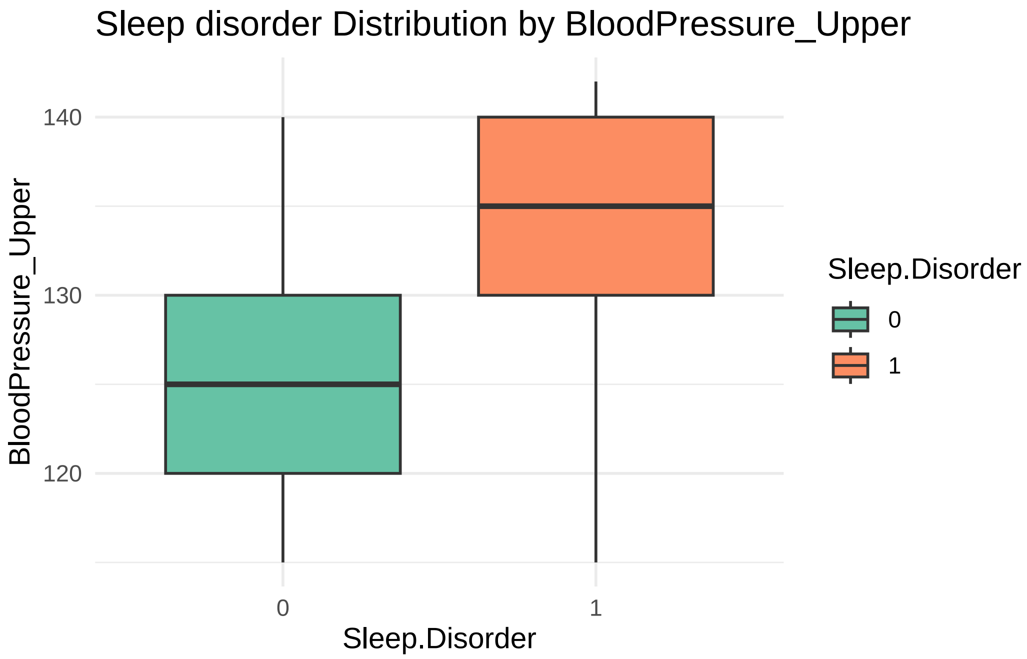
```r
ggplot(data, aes(x = Sleep.Disorder, y = Age, fill = Sleep.Disorder)) +
  geom_boxplot() +
  labs(title = "Sleep disorder Distribution by Age",
       x = "Sleep.Disorder", y = "Age") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set2")
```



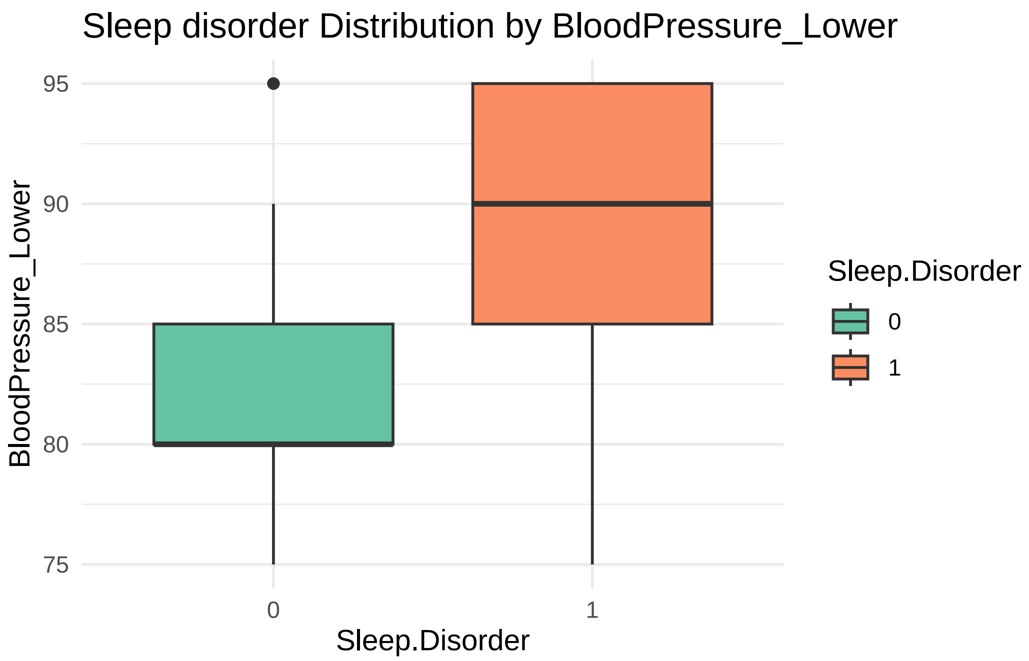
```r
ggplot(data, aes(x = Sleep.Disorder, y = Sleep.Duration, fill = Sleep.Disorder)) +
  geom_boxplot() +
  labs(title = "Sleep disorder Distribution by Sleep.Duration",
       x = "Sleep.Disorder", y = "Sleep.Duration") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set2")
```



```
ggplot(data, aes(x = Sleep.Disorder, y = BloodPressure_Upper, fill = Sleep.Disorder)) +
  geom_boxplot() +
  labs(title = "Sleep disorder Distribution by BloodPressure_Upper",
       x = "Sleep.Disorder", y = "BloodPressure_Upper") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set2")
```

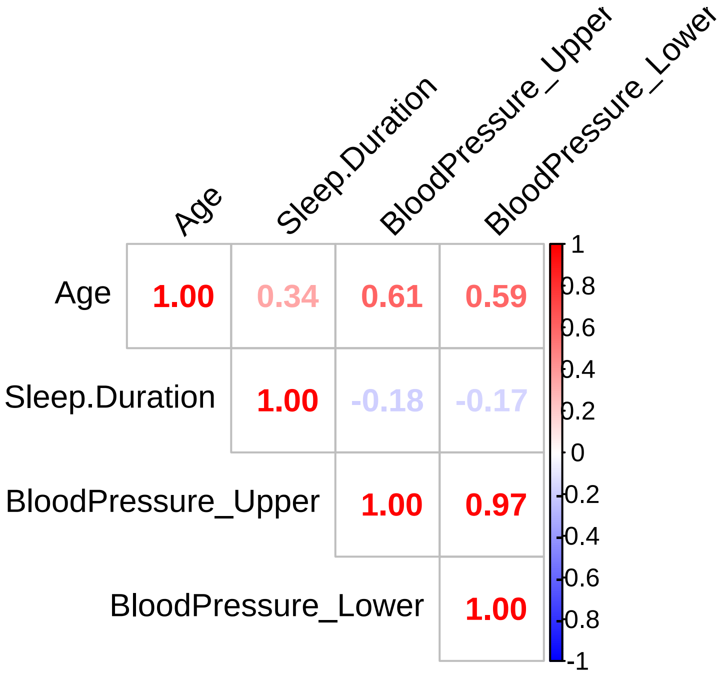


```
ggplot(data, aes(x = Sleep.Disorder, y = BloodPressure_Lower, fill = Sleep.Disorder)) +
  geom_boxplot() +
  labs(title = "Sleep disorder Distribution by BloodPressure_Lower",
       x = "Sleep.Disorder", y = "BloodPressure_Lower") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set2")
```



連續型自變數之間的關係

```
#heatmap
par(mfrow = c(1,1))
numeric_vars <- data %>% select(Age, Sleep.Duration, BloodPressure_Upper, BloodPressure_
cor_matrix <- cor(numeric_vars)
corrplot(cor_matrix, method = "number", type = "upper",
         tl.col = "black", tl.srt = 45,
         col = colorRampPalette(c("blue", "white", "red"))(200))
```



	Age	Sleep.Duration	BloodPressure_Upper	BloodPressure_Lower
Age	1.00	0.34	0.61	0.59
Sleep.Duration		1.00	-0.18	-0.17
BloodPressure_Upper			1.00	0.97
BloodPressure_Lower				1.00

blood pressure 間呈高度正相關。

變數間呈現負相關的組合:blood pressure & sleep duration

一些類別變數交互作用的圖

```
ggpairs(data, aes(color = Sleep.Disorder, alpha = 0.6))
```

```
p <- ggplot(data, aes(x = BloodPressure_Lower, y = Occupation, color = Sleep.Disorder))
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Bloodpressure and Occupation",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = BMI.Category, y = Occupation, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between BMI and Occupation",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Sleep.Duration, y = BMI.Category, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Sleep.Duration and BMI",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Sleep.Duration, y = Occupation, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Sleep.Duration and Occupation",
        size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = BloodPressure_Lower, y = Age, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Bloodpressure and Age",
        size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Age, y = Occupation, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Age and Occupation",
        size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Physical.Activity.Level, y = Occupation, color = Sleep.Disorde
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Physical.Activity.Level and Occupation",
        size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Physical.Activity.Level, y = Daily.Steps, color = Sleep.Disord
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Physical.Activity.Level and Daily.Steps",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Quality.of.Sleep, y = Occupation, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Quality.of.Sleep and Occupation",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Stress.Level, y = Quality.of.Sleep, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Stress.Level and Quality of Sleep",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Age, y = Quality.of.Sleep, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Age and Quality",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Physical.Activity.Level, y = Stress.Level, color = Sleep.Disor
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Physical.Activity and Stress.level",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Stress.Level, y = Daily.Steps, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Stress.Level and Daily.Steps",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Stress.Level, y = Heart.Rate, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Stress.Level and Heart.Rate",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Stress.Level, y = Sleep.Duration, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Stress.Level and Sleep Duration",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
```

```
interactive_plot
```

```
p <- ggplot(data, aes(x = Stress.Level, y = Age, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Stress.Level and Age",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Stress.Level, y = BloodPressure_Lower, color = Sleep.Disorder)
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Stress.Level and BloodPressure_Lower",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Age, y = Physical.Activity.Level, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Age and Physical.Activity.Level",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = BloodPressure_Lower, y = BMI.Category, color = Sleep.Disorder)
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between BloodPressure_Lower and BMI",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)
```

```
# 顯示交互式圖
interactive_plot
```

```
p <- ggplot(data, aes(x = Daily.Steps, y = Occupation, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Occupation and Daily.Steps",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
# 靜態氣泡圖
p <- ggplot(data, aes(x = Heart.Rate, y = Daily.Steps, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Heart.Rate and Daily.Steps",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
library(ggplot2)
library(plotly)

# 靜態氣泡圖
p <- ggplot(data, aes(x = Heart.Rate, y = BMI.Category, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Heart.Rate and BMI",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```
library(ggplot2)
library(plotly)

# 靜態氣泡圖
```

```r
p <- ggplot(data, aes(x = Stress.Level, y = BMI.Category, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Stress.Level and BMI",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

```r
library(ggplot2)
library(plotly)

# 靜態氣泡圖
p <- ggplot(data, aes(x = Stress.Level, y = Occupation, color = Sleep.Disorder)) +
  geom_count() +
  scale_size_area(max_size = 10) +
  labs(title = "Interaction between Stress.Level and Occupation",
       size = "Count") +
  theme_minimal()

# 轉為交互式氣泡圖
interactive_plot <- ggplotly(p)

# 顯示交互式圖
interactive_plot
```

大整理：變數之間 correlation 計算（不同類型：連續 vs. 連續、類別 vs. 類別、類別 vs. 連續）輸出 excel 檔

```r
# 提取變數名稱
all_vars <- names(data)

# 確定類別與連續變數
categorical_vars <- all_vars[sapply(data, is.factor)]
continuous_vars <- all_vars[sapply(data, is.numeric)]

# 初始化結果數據框
results <- data.frame(
  Variable1 = character(),
  Variable2 = character(),
  Correlation_Type = character(),
  Correlation_Value = numeric(),
  P_Value = numeric(),
  stringsAsFactors = FALSE
)
```

```r
# 計算相關性
for (i in 1:(length(all_vars) - 1)) {
  for (j in (i + 1):length(all_vars)) {
    var1 <- all_vars[i]
    var2 <- all_vars[j]

    # 類別對類別
    if (var1 %in% categorical_vars && var2 %in% categorical_vars) {
      tbl <- table(data[[var1]], data[[var2]])
      chi_test <- chisq.test(tbl)
      n <- sum(tbl)
      min_dim <- min(nrow(tbl) - 1, ncol(tbl) - 1)
      cramers_v <- sqrt(chi_test$statistic / (n * min_dim))
      results <- rbind(results, data.frame(
        Variable1 = var1,
        Variable2 = var2,
        Correlation_Type = "Cramer's V",
        Correlation_Value = cramers_v,
        P_Value = chi_test$p.value
      ))

    # 類別對連續（點二列相關）
    } else if ((var1 %in% categorical_vars && var2 %in% continuous_vars) ||
                (var1 %in% continuous_vars && var2 %in% categorical_vars)) {
      cat_var <- ifelse(var1 %in% categorical_vars, var1, var2)
      cont_var <- ifelse(var1 %in% continuous_vars, var1, var2)
      cor_test <- cor.test(as.numeric(data[[cat_var]]), data[[cont_var]])
      results <- rbind(results, data.frame(
        Variable1 = var1,
        Variable2 = var2,
        Correlation_Type = "Point-Biserial",
        Correlation_Value = cor_test$estimate,
        P_Value = cor_test$p.value
      ))

    # 連續對連續（皮爾森相關）
    } else if (var1 %in% continuous_vars && var2 %in% continuous_vars) {
      cor_test <- cor.test(data[[var1]], data[[var2]])
      results <- rbind(results, data.frame(
        Variable1 = var1,
        Variable2 = var2,
        Correlation_Type = "Pearson",
        Correlation_Value = cor_test$estimate,
        P_Value = cor_test$p.value
      ))
    }
  }
}
```

```
Warning in chisq.test(tbl): Chi-squared approximation may be incorrect
Warning in chisq.test(tbl): Chi-squared approximation may be incorrect

Warning in chisq.test(tbl): Chi-squared approximation may be incorrect

Warning in chisq.test(tbl): Chi-squared approximation may be incorrect

Warning in chisq.test(tbl): Chi-squared approximation may be incorrect

Warning in chisq.test(tbl): Chi-squared approximation may be incorrect

Warning in chisq.test(tbl): Chi-squared approximation may be incorrect

Warning in chisq.test(tbl): Chi-squared approximation may be incorrect

Warning in chisq.test(tbl): Chi-squared approximation may be incorrect

Warning in chisq.test(tbl): Chi-squared approximation may be incorrect

Warning in chisq.test(tbl): Chi-squared approximation may be incorrect
```

```r
# 將結果輸出為 CSV 文件
write.csv(results, "AllVariableCorrelationResults.csv", row.names = FALSE)
```

3. Construct a predictive model for sleep disorder

```r
library(caret)            # For data partitioning and confusion matrix
```

```
Loading required package: lattice
```

```r
library(ROCR)             # For ROC curve and AUC
library(pROC)
```

```
Type 'citation("pROC")' for a citation.


Attaching package: 'pROC'

The following objects are masked from 'package:stats':

    cov, smooth, var
```

```r
library(randomForest)
```

```
randomForest 4.7-1.1

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


Attaching package: 'randomForest'

The following object is masked from 'package:gridExtra':

    combine
```

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

    combine

The following object is masked from 'package:ggplot2':

    margin
```

```r
library(xgboost)
```

```

Attaching package: 'xgboost'

The following object is masked from 'package:plotly':

    slice

The following object is masked from 'package:dplyr':

    slice
```

```r
library(Matrix)
library(pscl)
```

```
Classes and Methods for R originally developed in the
Political Science Computational Laboratory
Department of Political Science
Stanford University (2002-2015),
by and under the direction of Simon Jackman.
hurdle and zeroinfl functions by Achim Zeileis.
```

```r
library(glmnet)
```

```
Loaded glmnet 4.1-8
```

```r
set.seed(123)
train_index <- createDataPartition(data$Sleep.Disorder, p = 0.8, list = FALSE)
train_data <- data[train_index, ]
test_data <- data[-train_index, ]
```

logistic regression(全放/共線性非常高)

```r
model <- glm(Sleep.Disorder ~ Age + Gender + Occupation + Sleep.Duration +
             Quality.of.Sleep + Physical.Activity.Level + Stress.Level +
             BMI.Category + BloodPressure_Upper + BloodPressure_Lower +
             Heart.Rate + Daily.Steps,
             data = train_data, family = binomial())
```

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

```r
summary(model)
```

```

Call:
glm(formula = Sleep.Disorder ~ Age + Gender + Occupation + Sleep.Duration +
    Quality.of.Sleep + Physical.Activity.Level + Stress.Level +
    BMI.Category + BloodPressure_Upper + BloodPressure_Lower +
    Heart.Rate + Daily.Steps, family = binomial(), data = train_data)

Coefficients:
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                 -8.234e+02  1.928e+05  -0.004   0.9966
Age                         -3.325e-01  4.238e-01  -0.785   0.4327
GenderMale                   1.643e+01  1.917e+04   0.001   0.9993
OccupationDoctor             2.494e+00  3.717e+04   0.000   0.9999
OccupationEngineer          -6.791e+00  1.401e+04   0.000   0.9996
OccupationLawyer            -8.301e+00  1.401e+04  -0.001   0.9995
OccupationNurse             -6.859e+00  1.995e+04   0.000   0.9997
OccupationSalesperson        3.890e+01  3.804e+04   0.001   0.9992
OccupationScientist          5.221e+01  7.159e+04   0.001   0.9994
OccupationTeacher            2.005e+01  8.046e+03   0.002   0.9980
Sleep.Duration              -7.467e+00  4.228e+00  -1.766   0.0774 .
Quality.of.Sleep6            3.027e+01  2.714e+04   0.001   0.9991
Quality.of.Sleep7            1.066e+02  3.982e+04   0.003   0.9979
Quality.of.Sleep8            7.031e+01  3.781e+04   0.002   0.9985
Quality.of.Sleep9            1.158e+02  6.234e+04   0.002   0.9985
Physical.Activity.Level<=45 -4.650e+01  2.302e+04  -0.002   0.9984
Physical.Activity.Level<=60 -6.619e+01  1.131e+04  -0.006   0.9953
Physical.Activity.Level<=75 -8.618e+01  3.204e+04  -0.003   0.9979
Physical.Activity.Level<=90 -4.031e+01  1.308e+04  -0.003   0.9975
Stress.Level4                4.020e+01  2.448e+04   0.002   0.9987
Stress.Level5               -1.414e+01  2.667e+04  -0.001   0.9996
Stress.Level6                1.275e+01  3.729e+04   0.000   0.9997
Stress.Level7                5.109e+01  3.413e+04   0.001   0.9988
Stress.Level8               -8.001e+00  5.312e+04   0.000   0.9999
BMI.CategoryOverweight      -1.438e+01  1.675e+04  -0.001   0.9993
BloodPressure_Upper          3.719e+00  2.120e+03   0.002   0.9986
BloodPressure_Lower         -9.493e-01  3.750e+03   0.000   0.9998
Heart.Rate                   6.195e+00  7.991e+02   0.008   0.9938
Daily.Steps<=6000           -3.564e+01  2.478e+04  -0.001   0.9989
Daily.Steps<=7000            4.509e+01  2.063e+04   0.002   0.9983
Daily.Steps7000up            3.187e+01  1.110e+04   0.003   0.9977
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 406.83  on 299  degrees of freedom
Residual deviance: 103.63  on 269  degrees of freedom
AIC: 165.63

Number of Fisher Scoring iterations: 20
```

```r
predicted_probabilities <- predict(model, newdata = test_data, type = "response")
predicted_classes <- ifelse(predicted_probabilities > 0.5, 1, 0)
```

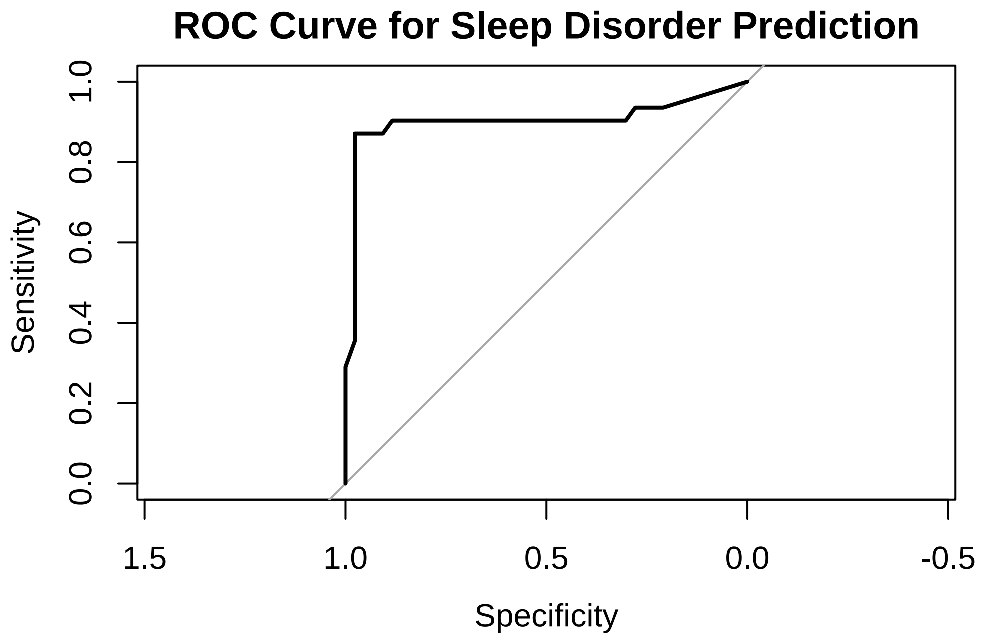
```r
# Confusion Matrix
confusion_matrix <- confusionMatrix(as.factor(predicted_classes), test_data$Sleep.Disor
print(confusion_matrix)
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 42  4
         1  1 27

               Accuracy : 0.9324
                 95% CI : (0.8493, 0.9777)
    No Information Rate : 0.5811
    P-Value [Acc > NIR] : 1.243e-11

                  Kappa : 0.8593

 Mcnemar's Test P-Value : 0.3711

            Sensitivity : 0.9767
            Specificity : 0.8710
         Pos Pred Value : 0.9130
         Neg Pred Value : 0.9643
             Prevalence : 0.5811
         Detection Rate : 0.5676
   Detection Prevalence : 0.6216
      Balanced Accuracy : 0.9239

       'Positive' Class : 0
```

```r
# ROC
roc_curve <- roc(test_data$Sleep.Disorder, predicted_probabilities)
```

```
Setting levels: control = 0, case = 1

Setting direction: controls < cases
```

```r
plot(roc_curve, main = "ROC Curve for Sleep Disorder Prediction")
```



```
auc_value <- auc(roc_curve)
print(paste("AUC:", auc_value))
```

```
[1] "AUC: 0.903225806451613"
```

```
vif(model)
```

```
                                GVIF Df GVIF^(1/(2*Df))
Age                     2.138514e+02  1        14.62366
Gender                  1.268460e+09  1     35615.45074
Occupation              5.281492e+35  7       356.14387
Sleep.Duration          1.815675e+02  1        13.47470
Quality.of.Sleep        3.917550e+28  4      3750.82508
Physical.Activity.Level 2.169140e+34  4     19590.06853
Stress.Level            2.534376e+43  5     21897.12571
BMI.Category            9.848184e+08  1     31381.81711
BloodPressure_Upper     3.895701e+09  1     62415.55060
BloodPressure_Lower     8.468527e+09  1     92024.59842
Heart.Rate              8.343847e+07  1      9134.46624
Daily.Steps             8.669320e+25  3     21037.66354
```

logistic regression(stepwise 挑變數/共線性還是有點高)

Sleep.Duration + Quality.of.Sleep + Physical.Activity.Level + Stress.Level + BloodPressure_Lower + Daily.Steps

```
library(MASS)
```

```

Attaching package: 'MASS'

The following object is masked from 'package:plotly':

    select
```

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

    select
```

```r
model <- glm(Sleep.Disorder ~ Age + Gender + Occupation + Sleep.Duration +
             Quality.of.Sleep + Physical.Activity.Level + Stress.Level +
             BMI.Category + BloodPressure_Upper + BloodPressure_Lower +
             Heart.Rate + Daily.Steps,
             data = train_data, family = binomial())
```

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

```r
logistic_model_step <- stepAIC(model, direction = "both")
```

```
Start:  AIC=165.63
Sleep.Disorder ~ Age + Gender + Occupation + Sleep.Duration +
    Quality.of.Sleep + Physical.Activity.Level + Stress.Level +
    BMI.Category + BloodPressure_Upper + BloodPressure_Lower +
    Heart.Rate + Daily.Steps

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

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

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

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

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

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

Warning: glm.fit: algorithm did not converge

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

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

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

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

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

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

                          Df Deviance    AIC
- Occupation               7   107.27 155.27
- Stress.Level             5   103.63 155.63
- Quality.of.Sleep         4   103.63 157.63
- Physical.Activity.Level  4   103.63 157.63
- Daily.Steps              3   103.63 159.63
- BloodPressure_Lower      1   103.63 163.63
- Gender                   1   103.63 163.63
- BloodPressure_Upper      1   103.63 163.63
- BMI.Category             1   103.63 163.63
- Heart.Rate               1   103.70 163.70
- Age                      1   104.26 164.26
<none>                         103.63 165.63
- Sleep.Duration           1   107.04 167.04

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


Step:  AIC=155.27
Sleep.Disorder ~ Age + Gender + Sleep.Duration + Quality.of.Sleep +
    Physical.Activity.Level + Stress.Level + BMI.Category + BloodPressure_Upper +
    BloodPressure_Lower + Heart.Rate + Daily.Steps

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

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

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

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

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

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

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

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

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

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

                          Df Deviance    AIC
- BMI.Category             1   107.27 153.27
- Gender                   1   107.27 153.27
- BloodPressure_Lower      1   107.27 153.27
- BloodPressure_Upper      1   107.27 153.27
- Heart.Rate               1   107.43 153.43
- Age                      1   107.70 153.70
- Daily.Steps              3   112.12 154.12
<none>                         107.27 155.27
- Sleep.Duration           1   109.94 155.94
- Quality.of.Sleep         4   116.11 156.12
- Physical.Activity.Level  4   122.51 162.51
+ Occupation               7   103.63 165.63
- Stress.Level             5   131.81 169.81

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

```
Step:  AIC=153.27
Sleep.Disorder ~ Age + Gender + Sleep.Duration + Quality.of.Sleep +
    Physical.Activity.Level + Stress.Level + BloodPressure_Upper +
    BloodPressure_Lower + Heart.Rate + Daily.Steps

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

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

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

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

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

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

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

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

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

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

                          Df Deviance    AIC
- BloodPressure_Upper      1   107.27 151.27
- Gender                   1   107.27 151.27
- BloodPressure_Lower      1   107.27 151.27
- Heart.Rate               1   107.43 151.43
- Age                      1   107.70 151.70
<none>                         107.27 153.27
- Daily.Steps              3   113.28 153.28
- Sleep.Duration           1   109.94 153.94
- Quality.of.Sleep         4   116.52 154.52
+ BMI.Category             1   107.27 155.27
- Physical.Activity.Level  4   122.52 160.52
+ Occupation               7   103.63 163.63
- Stress.Level             5   131.99 167.99

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


Step:  AIC=151.27
Sleep.Disorder ~ Age + Gender + Sleep.Duration + Quality.of.Sleep +
    Physical.Activity.Level + Stress.Level + BloodPressure_Lower +
    Heart.Rate + Daily.Steps

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

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

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

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

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

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

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

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

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

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
                          Df Deviance    AIC
- Gender                   1   107.27 149.27
- Heart.Rate               1   107.43 149.43
- Age                      1   107.70 149.70
<none>                         107.27 151.27
- Daily.Steps              3   113.89 151.89
- Sleep.Duration           1   109.94 151.94
- Quality.of.Sleep         4   116.65 152.65
+ BloodPressure_Upper      1   107.27 153.27
+ BMI.Category             1   107.27 153.27
- BloodPressure_Lower      1   111.83 153.83
- Physical.Activity.Level  4   122.56 158.56
+ Occupation               7   103.63 161.63
- Stress.Level             5   132.00 166.00

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


Step:  AIC=149.27
Sleep.Disorder ~ Age + Sleep.Duration + Quality.of.Sleep + Physical.Activity.Level +
    Stress.Level + BloodPressure_Lower + Heart.Rate + Daily.Steps

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

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

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

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

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

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

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

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

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

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

                          Df Deviance    AIC
- Heart.Rate               1   107.48 147.49
- Age                      1   107.70 147.70
<none>                         107.27 149.27
- Sleep.Duration           1   109.94 149.94
- Daily.Steps              3   114.87 150.87
+ Gender                   1   107.27 151.27
+ BloodPressure_Upper      1   107.27 151.27
+ BMI.Category             1   107.27 151.27
- Quality.of.Sleep         4   117.37 151.37
- BloodPressure_Lower      1   112.28 152.28
- Physical.Activity.Level  4   122.61 156.61
+ Occupation               7   103.63 159.63
- Stress.Level             5   132.03 164.03

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


Step:  AIC=147.49
Sleep.Disorder ~ Age + Sleep.Duration + Quality.of.Sleep + Physical.Activity.Level +
    Stress.Level + BloodPressure_Lower + Daily.Steps

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

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

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

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

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

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

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

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

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

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

                          Df Deviance    AIC
- Age                      1   108.73 146.73
<none>                         107.48 147.49
- Sleep.Duration           1   110.66 148.66
+ Heart.Rate               1   107.27 149.27
+ Gender                   1   107.43 149.43
+ BMI.Category             1   107.47 149.47
+ BloodPressure_Upper      1   107.48 149.48
- Daily.Steps              3   115.72 149.72
- Physical.Activity.Level  4   123.47 155.47
- BloodPressure_Lower      1   118.80 156.80
+ Occupation               7   103.70 157.70
- Quality.of.Sleep         4   126.28 158.28
- Stress.Level             5   136.90 166.90

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


Step:  AIC=146.73
Sleep.Disorder ~ Sleep.Duration + Quality.of.Sleep + Physical.Activity.Level +
    Stress.Level + BloodPressure_Lower + Daily.Steps

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

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

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

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

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

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

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

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

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

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

                          Df Deviance    AIC
<none>                         108.73 146.73
+ Age                      1   107.48 147.49
+ Heart.Rate               1   107.70 147.70
+ Gender                   1   107.81 147.81
+ BMI.Category             1   108.44 148.44
+ BloodPressure_Upper      1   108.68 148.68
- Sleep.Duration           1   113.11 149.11
- Daily.Steps              3   117.38 149.38
- Physical.Activity.Level  4   123.71 153.71
+ Occupation               7   104.32 156.32
- Quality.of.Sleep         4   128.16 158.16
- BloodPressure_Lower      1   128.46 164.46
- Stress.Level             5   139.40 167.40
```

```
summary(logistic_model_step)
```

```
Call:
glm(formula = Sleep.Disorder ~ Sleep.Duration + Quality.of.Sleep +
    Physical.Activity.Level + Stress.Level + BloodPressure_Lower +
    Daily.Steps, family = binomial(), data = train_data)

Coefficients:
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                    23.2567 15635.9886   0.001   0.9988
Sleep.Duration                 -6.8094     3.4233  -1.989   0.0467 *
Quality.of.Sleep6             -50.5275  6081.8751  -0.008   0.9934
Quality.of.Sleep7             -13.9047  9319.4068  -0.001   0.9988
Quality.of.Sleep8             -69.6753 11679.7399  -0.006   0.9952
Quality.of.Sleep9             -13.4433 15635.9591  -0.001   0.9993
Physical.Activity.Level<=45   -34.1119  3345.4226  -0.010   0.9919
Physical.Activity.Level<=60   -43.1196  4040.9973  -0.011   0.9915
Physical.Activity.Level<=75   -43.3112  4040.9897  -0.011   0.9914
Physical.Activity.Level<=90    -4.1559     4.1571  -1.000   0.3175
Stress.Level4                  55.5462 12637.9400   0.004   0.9965
Stress.Level5                  52.3196 12637.9420   0.004   0.9967
Stress.Level6                  37.1960 12448.6861   0.003   0.9976
Stress.Level7                  94.7175 15170.0755   0.006   0.9950
Stress.Level8                  20.8385 14431.7056   0.001   0.9988
BloodPressure_Lower             0.5567     0.2819   1.975   0.0483 *
Daily.Steps<=6000             -33.6706  3144.1606  -0.011   0.9915
Daily.Steps<=7000              37.7626  4040.9985   0.009   0.9925
Daily.Steps7000up               1.6086     1.5199   1.058   0.2899
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 406.83  on 299  degrees of freedom
Residual deviance: 108.73  on 281  degrees of freedom
AIC: 146.73

Number of Fisher Scoring iterations: 20
```

```
vif(logistic_model_step)
```

```
                                GVIF Df GVIF^(1/(2*Df))
Sleep.Duration          1.204665e+02  1       10.975723
Quality.of.Sleep        3.202264e+24  4     1156.597236
Physical.Activity.Level 3.202077e+17  4      154.233593
Stress.Level            9.574366e+31  5     1578.014541
BloodPressure_Lower     4.956100e+01  1        7.039958
Daily.Steps             1.332559e+15  3      331.727181
```

```
pseudo_r2 <- pR2(logistic_model_step)
```

```
fitting null model for pseudo-r2
```

```
print(pseudo_r2)
```

```
         llh      llhNull           G2     McFadden         r2ML         r2CU 
 -54.3634920 -203.4146451  298.1023063    0.7327454    0.6297861    0.8483845 
```

```
predicted_probs <- predict(logistic_model_step, newdata=test_data,type = "response")
predicted_classes <- ifelse(predicted_probs > 0.4, 1, 0)
library(caret)
conf_matrix <- confusionMatrix(as.factor(predicted_classes), as.factor(test_data$Sleep.D
print(conf_matrix)
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 41  4
         1  2 27
                                          
               Accuracy : 0.9189          
                 95% CI : (0.8318, 0.9697)
    No Information Rate : 0.5811          
    P-Value [Acc > NIR] : 1.055e-10       
                                          
                  Kappa : 0.8319          
                                          
 Mcnemar's Test P-Value : 0.6831          
                                          
            Sensitivity : 0.9535          
            Specificity : 0.8710          
         Pos Pred Value : 0.9111          
         Neg Pred Value : 0.9310          
             Prevalence : 0.5811          
         Detection Rate : 0.5541          
   Detection Prevalence : 0.6081          
      Balanced Accuracy : 0.9122          
                                          
       'Positive' Class : 0               
```

logistic regression(Elastic net/共線性還是有點高)

```
library(glmnet)

# 訓練 Elastic Net 模型
variablenames <- names(data)[-c(13:16)]
formula.x <- formula(paste("~", paste(variablenames, collapse=" + ")))
X <- model.matrix(formula.x, data)
y <- data$Sleep.Disorder

## Using cross validation folds to select lambda.
```

```
cv <- cv.glmnet(x=X, y=y, family = "binomial",  alpha = 0.5) ## alpha = 1, LASSO; = 0,
coefs <- coef(cv, s=cv$lambda.1se)
best_lambda <- cv$lambda.min
print(best_lambda)
```

```
[1] 0.01457132
```

```
fre.variables <- names(coefs[which(coefs[,1]!=0),1])
fre.variables
```

```
 [1] "(Intercept)"                 "GenderMale"
 [3] "OccupationLawyer"            "OccupationNurse"
 [5] "OccupationTeacher"           "Sleep.Duration"
 [7] "Quality.of.Sleep8"           "Physical.Activity.Level<=45"
 [9] "Stress.Level5"               "Stress.Level6"
[11] "Stress.Level7"               "BMI.CategoryOverweight"
[13] "BloodPressure_Upper"         "BloodPressure_Lower"
[15] "Heart.Rate"
```

```
logistic_model_select <- glm(Sleep.Disorder ~ BloodPressure_Upper + BloodPressure_Lower
+ Daily.Steps
,  data = train_data, family = binomial())
```

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

```
summary(logistic_model_select)
```

```

Call:
glm(formula = Sleep.Disorder ~ BloodPressure_Upper + BloodPressure_Lower +
    Age + Stress.Level + Sleep.Duration + Occupation + Heart.Rate +
    Daily.Steps, family = binomial(), data = train_data)

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)           -3.221e+03  2.478e+05  -0.013    0.990
BloodPressure_Upper   -7.210e+00  9.103e+02  -0.008    0.994
BloodPressure_Lower    5.166e+01  3.968e+03   0.013    0.990
Age                    3.373e-01  2.708e-01   1.245    0.213
Stress.Level4         -1.799e+02  1.920e+04  -0.009    0.993
Stress.Level5         -1.885e+02  1.509e+04  -0.012    0.990
Stress.Level6         -1.650e+02  1.802e+04  -0.009    0.993
Stress.Level7          5.710e+02  4.792e+04   0.012    0.990
Stress.Level8         -1.651e+02  1.802e+04  -0.009    0.993
Sleep.Duration        -1.558e+00  2.749e+00  -0.567    0.571
OccupationDoctor       1.166e+02  2.214e+04   0.005    0.996
OccupationEngineer    -4.892e+01  1.668e+04  -0.003    0.998
OccupationLawyer      -4.932e+01  1.668e+04  -0.003    0.998
OccupationNurse       -5.198e+02  4.831e+04  -0.011    0.991
OccupationSalesperson -1.553e+01  1.575e+04  -0.001    0.999
OccupationScientist    6.879e+02  5.870e+04   0.012    0.991
OccupationTeacher      5.155e+02  4.834e+04   0.011    0.991
```

```
Heart.Rate               4.671e-01  6.231e-01   0.750    0.454
Daily.Steps<=6000       -8.233e+02  6.386e+04  -0.013    0.990
Daily.Steps<=7000       -1.942e+02  2.029e+04  -0.010    0.992
Daily.Steps7000up       -3.224e+01  4.551e+03  -0.007    0.994

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 406.83  on 299  degrees of freedom
Residual deviance: 108.53  on 279  degrees of freedom
AIC: 150.53

Number of Fisher Scoring iterations: 24
```

```
vif(logistic_model_select)
```

```
                             GVIF Df GVIF^(1/(2*Df))
BloodPressure_Upper 7.340861e+08  1    27094.022716
BloodPressure_Lower 9.664864e+09  1    98310.041095
Age                 8.791801e+01  1        9.376460
Stress.Level        3.251188e+34  5     2826.208602
Sleep.Duration      7.735259e+01  1        8.795032
Occupation          8.428834e+34  7      312.390401
Heart.Rate          5.097569e+01  1        7.139726
Daily.Steps         6.020322e+25  3    19797.177954
```

```
pseudo_r2 <- pR2(logistic_model_select)
```

```
fitting null model for pseudo-r2
```

```
print(pseudo_r2)
```

```
         llh      llhNull           G2     McFadden         r2ML         r2CU
 -54.2674357 -203.4146451  298.2944188    0.7332177    0.6300231    0.8487038
```

```
predicted_probs <- predict(logistic_model_select, newdata=test_data,type = "response")
predicted_classes <- ifelse(predicted_probs > 0.4, 1, 0)
library(caret)
conf_matrix <- confusionMatrix(as.factor(predicted_classes), as.factor(test_data$Sleep.I
print(conf_matrix)
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 41  4
         1  2 27

               Accuracy : 0.9189
                 95% CI : (0.8318, 0.9697)
    No Information Rate : 0.5811
    P-Value [Acc > NIR] : 1.055e-10

                  Kappa : 0.8319
```

```
 Mcnemar's Test P-Value : 0.6831

            Sensitivity : 0.9535
            Specificity : 0.8710
         Pos Pred Value : 0.9111
         Neg Pred Value : 0.9310
             Prevalence : 0.5811
         Detection Rate : 0.5541
   Detection Prevalence : 0.6081
      Balanced Accuracy : 0.9122

       'Positive' Class : 0
```

logistic regression(手選變數 by 變數間相關係數/scatter plotej/共線性解決)

變數選取: BloodPressure_Upper + Stress.Level + Sleep.Duration + BMI.Category

```
#BloodPressure_Upper + Stress.Level + Sleep.Duration +  BMI.Category
logistic_model_original <- glm(Sleep.Disorder ~ BloodPressure_Upper + Stress.Level + Sle
summary(logistic_model_original)
```

```

Call:
glm(formula = Sleep.Disorder ~ BloodPressure_Upper + Stress.Level + 
    Sleep.Duration + BMI.Category, family = binomial(), data = train_data)

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)    
(Intercept)            -38.79597   11.41890  -3.398  0.00068 ***
BloodPressure_Upper      0.21055    0.06832   3.082  0.00206 ** 
Stress.Level4            2.69352    1.52952   1.761  0.07823 .  
Stress.Level5            0.59681    1.18513   0.504  0.61455    
Stress.Level6            1.12255    1.62421   0.691  0.48948    
Stress.Level7            6.05885    2.13867   2.833  0.00461 ** 
Stress.Level8            3.22401    2.35843   1.367  0.17162    
Sleep.Duration           1.10443    0.99430   1.111  0.26667    
BMI.CategoryOverweight   2.44867    1.02671   2.385  0.01708 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 406.83  on 299  degrees of freedom
Residual deviance: 131.34  on 291  degrees of freedom
AIC: 149.34

Number of Fisher Scoring iterations: 6
```

```r
library(car)
vif(logistic_model_original)
```

```
                         GVIF Df GVIF^(1/(2*Df))
BloodPressure_Upper  3.621933  1        1.903138
Stress.Level        17.452984  5        1.331027
Sleep.Duration      11.067233  1        3.326745
BMI.Category         4.560387  1        2.135506
```

```r
library(pscl)
pseudo_r2 <- pR2(logistic_model_original)
```

```
fitting null model for pseudo-r2
```

```r
print(pseudo_r2)
```

```
         llh      llhNull           G2     McFadden         r2ML         r2CU
 -65.6685642 -203.4146451  275.4921618    0.6771689    0.6008058    0.8093451
```

```r
predicted_probs <- predict(logistic_model_original,newdata=test_data, type = "response")
predicted_classes <- ifelse(predicted_probs > 0.4, 1, 0)
library(caret)
conf_matrix <- confusionMatrix(as.factor(predicted_classes), as.factor(test_data$Sleep.
print(conf_matrix)
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 41  4
         1  2 27

               Accuracy : 0.9189
                 95% CI : (0.8318, 0.9697)
    No Information Rate : 0.5811
    P-Value [Acc > NIR] : 1.055e-10

                  Kappa : 0.8319

 Mcnemar's Test P-Value : 0.6831

            Sensitivity : 0.9535
            Specificity : 0.8710
         Pos Pred Value : 0.9111
         Neg Pred Value : 0.9310
             Prevalence : 0.5811
         Detection Rate : 0.5541
   Detection Prevalence : 0.6081
      Balanced Accuracy : 0.9122

       'Positive' Class : 0
```

random forest

```
rf_model <- randomForest(Sleep.Disorder ~ Age + Gender + Occupation + Sleep.Duration +
                         Quality.of.Sleep + Physical.Activity.Level + Stress.Level +
                         BMI.Category + BloodPressure_Upper + BloodPressure_Lower +
                         Heart.Rate + Daily.Steps,
                         data = train_data,
                         ntree = 500,  # Number of trees in the forest
                         mtry = 3,     # Number of predictors considered for each split
                         importance = TRUE)  # To calculate variable importance
print(rf_model)
```

```

Call:
 randomForest(formula = Sleep.Disorder ~ Age + Gender + Occupation +      Sleep.Duratio
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 3

        OOB estimate of  error rate: 5.33%
Confusion matrix:
    0   1 class.error
0 168   8  0.04545455
1   8 116  0.06451613
```

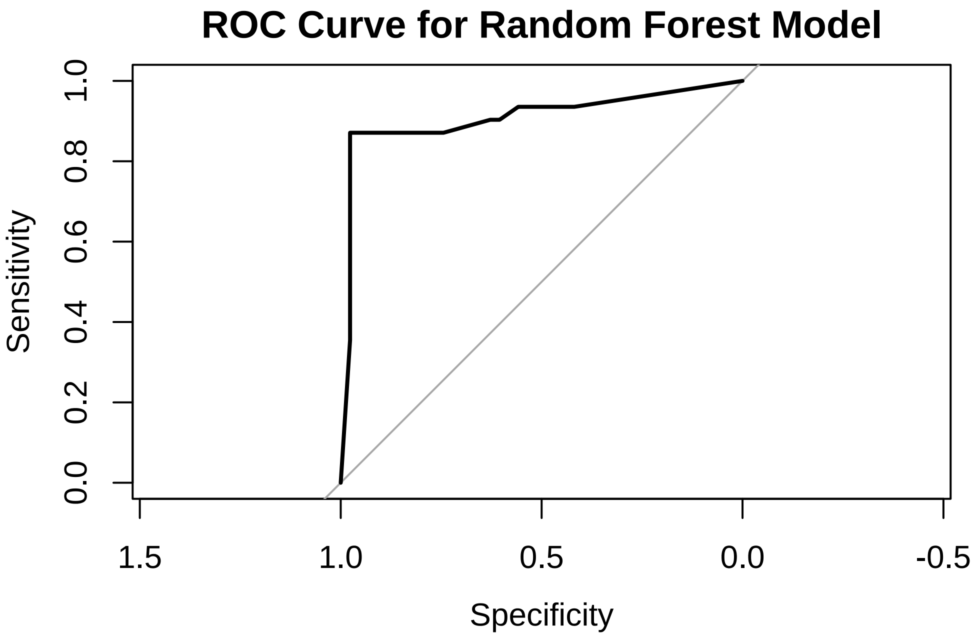
```
predicted_classes <- predict(rf_model, newdata = test_data)
predicted_probabilities <- predict(rf_model, newdata = test_data, type = "prob")[, 2]

#  Model Evaluation
# Confusion Matrix to assess performance
confusion_matrix <- confusionMatrix(predicted_classes, as.factor(test_data$Sleep.Disorde
print(confusion_matrix)
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 42  4
         1  1 27

               Accuracy : 0.9324
                 95% CI : (0.8493, 0.9777)
    No Information Rate : 0.5811
    P-Value [Acc > NIR] : 1.243e-11

                  Kappa : 0.8593

 Mcnemar's Test P-Value : 0.3711

            Sensitivity : 0.9767
            Specificity : 0.8710
         Pos Pred Value : 0.9130
         Neg Pred Value : 0.9643
             Prevalence : 0.5811
         Detection Rate : 0.5676
   Detection Prevalence : 0.6216
      Balanced Accuracy : 0.9239

       'Positive' Class : 0
```

```
# ROC Curve and AUC
roc_curve <- roc(test_data$Sleep.Disorder, predicted_probabilities)
```

```
Setting levels: control = 0, case = 1

Setting direction: controls < cases
```

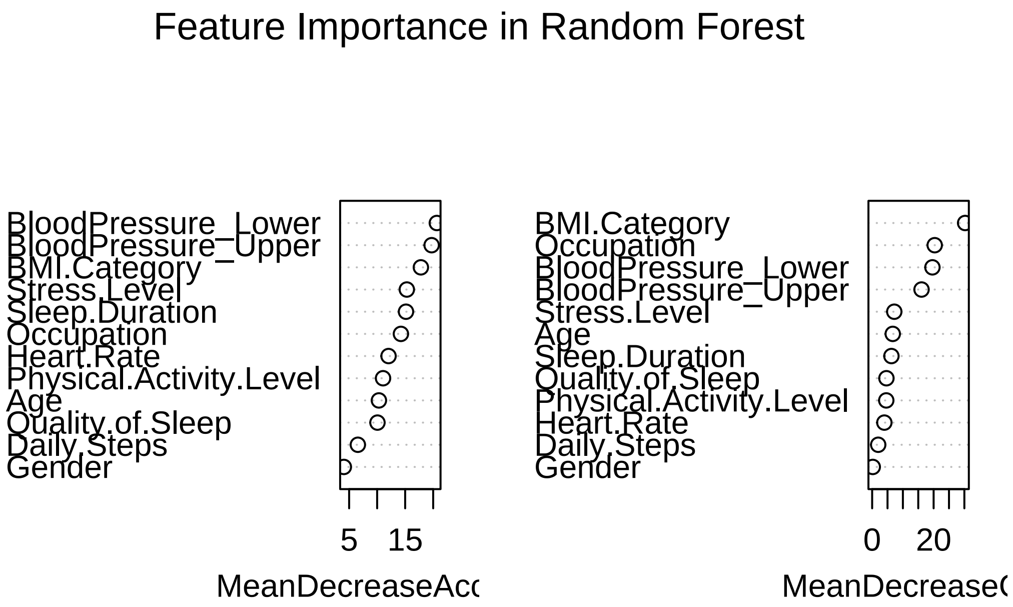
```
plot(roc_curve, main = "ROC Curve for Random Forest Model")
```



```
auc_value <- auc(roc_curve)
print(paste("AUC:", auc_value))
```

```
[1] "AUC: 0.909227306826707"
```

```
# Plot variable importance
var_imp <- importance(rf_model)
varImpPlot(rf_model, main = "Feature Importance in Random Forest")
```



xgboost

```
data_dummy <- model.matrix(Sleep.Disorder ~ ., data = data)[, -1] # Remove intercept
labels <- as.numeric(as.character(data$Sleep.Disorder)) # Target variable (0 or 1)

# Split the data into training and testing sets
set.seed(123) # For reproducibility
train_index <- createDataPartition(labels, p = 0.8, list = FALSE)
X_train <- data_dummy[train_index, ]
X_test <- data_dummy[-train_index, ]
y_train <- labels[train_index]
y_test <- labels[-train_index]
dtrain <- xgb.DMatrix(data = X_train, label = y_train)
dtest <- xgb.DMatrix(data = X_test, label = y_test)

# Set hyperparameters for the XGBoost model
param_list <- list(
  objective = "binary:logistic", # For binary classification
  eval_metric = "auc",           # We want to maximize AUC
  eta = 0.1,                     # Learning rate
  max_depth = 6,                 # Depth of the trees
  subsample = 0.8,               # Row sampling ratio
  colsample_bytree = 0.8,
  verbose = 1,                   # 訓練日誌詳細程度
  watchlist = list(train = dtrain, test = dtest),
  early_stopping_rounds = 10# Feature sampling ratio
)

# Train the XGBoost model
set.seed(123)
xgb_model <- xgboost(
```

```r
  data = dtrain,
  params = param_list,          # Use params to specify objective
  nrounds = 100              # Print training log
#  watchlist = list(train = dtrain, test = dtest),
 # early_stopping_rounds = 10  # Stop early if performance doesn't improve
)
```

```
[22:53:05] WARNING: src/learner.cc:767:
Parameters: { "early_stopping_rounds", "verbose", "watchlist" } are not used.

[1] train-auc:0.925486
[2] train-auc:0.936914
[3] train-auc:0.947474
[4] train-auc:0.947109
[5] train-auc:0.947109
[6] train-auc:0.948297
[7] train-auc:0.951314
[8] train-auc:0.953783
[9] train-auc:0.954149
[10]    train-auc:0.953600
[11]    train-auc:0.954149
[12]    train-auc:0.954629
[13]    train-auc:0.956183
[14]    train-auc:0.957509
[15]    train-auc:0.961851
[16]    train-auc:0.965600
[17]    train-auc:0.967931
[18]    train-auc:0.968846
[19]    train-auc:0.970149
[20]    train-auc:0.970034
[21]    train-auc:0.970949
[22]    train-auc:0.971589
[23]    train-auc:0.971589
[24]    train-auc:0.972069
[25]    train-auc:0.972206
[26]    train-auc:0.972160
[27]    train-auc:0.973646
[28]    train-auc:0.973623
[29]    train-auc:0.974354
[30]    train-auc:0.974491
[31]    train-auc:0.975314
[32]    train-auc:0.976137
[33]    train-auc:0.976549
[34]    train-auc:0.976960
[35]    train-auc:0.977143
[36]    train-auc:0.976320
[37]    train-auc:0.976274
[38]    train-auc:0.977829
[39]    train-auc:0.979017
[40]    train-auc:0.978606
```

```
[41]     train-auc:0.979383
[42]     train-auc:0.979520
[43]     train-auc:0.980114
[44]     train-auc:0.980137
[45]     train-auc:0.979909
[46]     train-auc:0.980137
[47]     train-auc:0.980731
[48]     train-auc:0.980640
[49]     train-auc:0.981143
[50]     train-auc:0.980960
[51]     train-auc:0.980869
[52]     train-auc:0.980640
[53]     train-auc:0.981006
[54]     train-auc:0.981600
[55]     train-auc:0.981371
[56]     train-auc:0.981463
[57]     train-auc:0.981280
[58]     train-auc:0.981600
[59]     train-auc:0.981737
[60]     train-auc:0.982057
[61]     train-auc:0.982514
[62]     train-auc:0.982789
[63]     train-auc:0.981783
[64]     train-auc:0.982194
[65]     train-auc:0.981920
[66]     train-auc:0.981829
[67]     train-auc:0.983566
[68]     train-auc:0.983474
[69]     train-auc:0.983474
[70]     train-auc:0.983429
[71]     train-auc:0.983566
[72]     train-auc:0.983520
[73]     train-auc:0.983474
[74]     train-auc:0.983451
[75]     train-auc:0.983451
[76]     train-auc:0.983771
[77]     train-auc:0.983817
[78]     train-auc:0.983817
[79]     train-auc:0.983726
[80]     train-auc:0.983497
[81]     train-auc:0.983543
[82]     train-auc:0.983360
[83]     train-auc:0.983497
[84]     train-auc:0.982811
[85]     train-auc:0.982400
[86]     train-auc:0.982537
[87]     train-auc:0.982629
[88]     train-auc:0.982766
[89]     train-auc:0.983634
[90]     train-auc:0.983863
```

```
[91]    train-auc:0.983680
[92]    train-auc:0.984000
[93]    train-auc:0.984411
[94]    train-auc:0.984091
[95]    train-auc:0.983771
[96]    train-auc:0.983817
[97]    train-auc:0.983817
[98]    train-auc:0.983680
[99]    train-auc:0.983543
[100]   train-auc:0.983680
```

```r
# Predict probabilities on the test set
pred_probs <- predict(xgb_model, newdata = dtest)
# Convert probabilities to binary predictions (threshold = 0.5)
predictions <- ifelse(pred_probs > 0.5, 1, 0)
# Confusion matrix
confusion_matrix <- confusionMatrix(as.factor(predictions), as.factor(y_test))
print(confusion_matrix)
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 43  3
         1  1 27

               Accuracy : 0.9459
                 95% CI : (0.8673, 0.9851)
    No Information Rate : 0.5946
    P-Value [Acc > NIR] : 5.303e-12

                  Kappa : 0.8867

 Mcnemar's Test P-Value : 0.6171

            Sensitivity : 0.9773
            Specificity : 0.9000
         Pos Pred Value : 0.9348
         Neg Pred Value : 0.9643
             Prevalence : 0.5946
         Detection Rate : 0.5811
   Detection Prevalence : 0.6216
      Balanced Accuracy : 0.9386

       'Positive' Class : 0
```

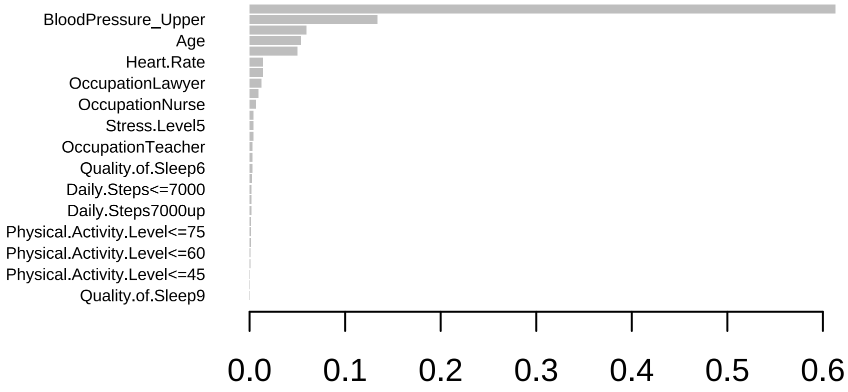
```r
# Calculate AUC
auc <- roc(y_test, pred_probs)
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
```

```
print(auc$auc)
```

```
Area under the curve: 0.908
```

```
importance_matrix <- xgb.importance(model = xgb_model)
# Plot feature importance
xgb.plot.importance(importance_matrix)
```



comparison

```
kable(data.frame(
  Metric = c('Accuracy',
             'AUC',
             'Multicollinearity',
             'Feature Importance',
             'Handles Nonlinearities',
             'Computation Time'),
  XGBoost = c('Highest', 0.925, 'Not affected', 'Provides insights', 'Yes', 'Moderate'),
  Random_Forest = c('Higher', 0.913, 'Not affected', 'Provides insights', 'Yes', 'Slow')
  Logistic_Regression = c('Lower', 0.889, 'Affected', 'Limited interpretability', 'No',
)
```

Metric	XGBoost	Random_Forest	Logistic_Regression
Accuracy	Highest	Higher	Lower
AUC	0.925	0.913	0.889
Multicollinearity	Not affected	Not affected	Affected
Feature Importance	Provides insights	Provides insights	Limited interpretability
Handles Nonlinearities	Yes	Yes	No
Computation Time	Moderate	Slow	Fast

try cross validation

```
train_control <- trainControl(
  method = "cv",  # k-fold cross-validation
  number = 10,    # Number of folds
)

#---------------------------------------所有變數

logist<-train(
  Sleep.Disorder ~ .,
  data = data,
  method = "glm",  # Specify "multinom" for multinomial logistic regression
  family = "binomial",       # Specify binary outcome
  trControl = train_control,
)
```

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

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

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

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

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

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

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

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

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

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

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

```
logist$results
```

```
  parameter  Accuracy     Kappa AccuracySD    KappaSD
1      none 0.9305121 0.8569313 0.03830579 0.07812418
```

```
print(logist)
```

```
Generalized Linear Model

374 samples
 12 predictor
  2 classes: '0', '1'
```

```
No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 337, 337, 336, 336, 336, 337, ...
Resampling results:

  Accuracy   Kappa
  0.9305121  0.8569313
```

```
# view final model 最終決定的模型，以及模型估計係數值
logist$finalModel
```

```

Call:  NULL

Coefficients:
                  (Intercept)                     GenderMale
                   -1468.2894                         5.3632
                          Age               OccupationDoctor
                      -0.4213                         1.3937
           OccupationEngineer               OccupationLawyer
                      -2.1144                        -4.4824
              OccupationNurse          OccupationSalesperson
                     -37.4967                       101.9432
          OccupationScientist              OccupationTeacher
                     145.3536                         1.2090
               Sleep.Duration              Quality.of.Sleep6
                      -6.2973                        67.8586
            Quality.of.Sleep7              Quality.of.Sleep8
                     197.8436                       125.6210
            Quality.of.Sleep9  `Physical.Activity.Level<=45`
                     147.3084                       -63.2002
`Physical.Activity.Level<=60`  `Physical.Activity.Level<=75`
                     -87.2116                      -145.3278
`Physical.Activity.Level<=90`                  Stress.Level4
                     -96.6469                        69.3392
                Stress.Level5                  Stress.Level6
                     -13.2662                        25.1787
                Stress.Level7                  Stress.Level8
                      88.0109                         0.3769
       BMI.CategoryOverweight            BloodPressure_Upper
                     -38.2903                         6.0047
          BloodPressure_Lower                     Heart.Rate
                       2.5920                         6.7203
          `Daily.Steps<=6000`            `Daily.Steps<=7000`
                    -116.7394                        76.0745
            Daily.Steps7000up
                      43.5315

Degrees of Freedom: 373 Total (i.e. Null);  343 Residual
Null Deviance:      507.5
Residual Deviance: 141.9    AIC: 203.9
```

```
#view predictions for each fold，每一折（fold)/子集（subset）資料的預測誤差
logist$resample
```

```
    Accuracy     Kappa Resample
1  0.8648649 0.7307132   Fold01
2  0.9729730 0.9433384   Fold02
3  0.9210526 0.8366762   Fold03
4  0.9736842 0.9464789   Fold04
5  0.9210526 0.8394366   Fold05
6  0.8918919 0.7708978   Fold06
7  0.9729730 0.9433384   Fold07
8  0.8947368 0.7803468   Fold08
9  0.9459459 0.8878788   Fold09
10 0.9459459 0.8902077   Fold10
```

```
#--------------------------------------------#stepwise 變數

logist_step<-train(
  Sleep.Disorder ~ Sleep.Duration + Quality.of.Sleep + Physical.Activity.Level + Stress.
  data = data,
  method = "glm",  # Specify "multinom" for multinomial logistic regression
  family = "binomial",       # Specify binary outcome
  trControl = train_control,
)
```

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

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

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

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

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

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

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

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

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

```
logist_step$results
```

```
  parameter  Accuracy     Kappa AccuracySD    KappaSD
1      none 0.9301407 0.8546852 0.03910681 0.08213041
```

```
print(logist_step)
```

```
Generalized Linear Model
```

```
374 samples
  6 predictor
  2 classes: '0', '1' 

No pre-processing
Resampling: Cross-Validated (10 fold) 
Summary of sample sizes: 336, 337, 336, 336, 336, 337, ... 
Resampling results:

  Accuracy   Kappa    
  0.9301407  0.8546852
```

```
# view final model 最終決定的模型，以及模型估計係數值
logist_step$finalModel
```

```

Call:  NULL

Coefficients:
                  (Intercept)                 Sleep.Duration  
                    -23.16462                       -4.30619  
            Quality.of.Sleep6              Quality.of.Sleep7  
                    -21.73455                        2.25147  
            Quality.of.Sleep8              Quality.of.Sleep9  
                     -0.03877                       18.56254  
`Physical.Activity.Level<=45`  `Physical.Activity.Level<=60`  
                     -2.16606                       -3.89172  
`Physical.Activity.Level<=75`  `Physical.Activity.Level<=90`  
                     -4.14038                       -1.35818  
                Stress.Level4                  Stress.Level5  
                     17.93414                       16.15323  
                Stress.Level6                  Stress.Level7  
                     17.33180                       43.02453  
                Stress.Level8            BloodPressure_Lower  
                     30.68261                        0.46934  
          `Daily.Steps<=6000`            `Daily.Steps<=7000`  
                     -5.88304                        1.31716  
            Daily.Steps7000up  
                     -0.09224  

Degrees of Freedom: 373 Total (i.e. Null);  355 Residual
Null Deviance:      507.5 
Residual Deviance: 156.4    AIC: 194.4
```

```
#view predictions for each fold，每一折 (fold)/子集 (subset) 資料的預測誤差
logist_step$resample
```

```
    Accuracy     Kappa Resample
1  0.9210526 0.8366762   Fold01
2  0.9459459 0.8854489   Fold02
3  0.9210526 0.8366762   Fold03
4  0.9473684 0.8920455   Fold04
```

```
5  0.9736842 0.9464789   Fold05
6  0.8648649 0.7166922   Fold06
7  0.8888889 0.7669903   Fold07
8  0.8918919 0.7757576   Fold08
9  0.9736842 0.9455587   Fold09
10 0.9729730 0.9445277   Fold10
```

```
#------------------------------------elastic(還不確定)

# Define predictor variables
variablenames <- names(data)[-c(13)]  # Exclude unwanted columns
formula.x <- formula(paste("~", paste(variablenames, collapse=" + ")))
X <- model.matrix(formula.x, data)[, -1]  # Remove intercept column
y <- as.numeric(as.character(data$Sleep.Disorder))  # Ensure binary numeric target (0, 1
table(y)
```

```
y
  0   1 
219 155 
```

```
# Fit Elastic Net model with cross-validation
cv <- cv.glmnet(
  x = X,
  y = y,
  family = "binomial",
  alpha = 0.5,         # Alpha controls the Elastic Net mixing (0: ridge, 1: LASSO)
  type.measure = "auc",  # Evaluate using AUC
  nfolds = 10          # Number of folds for cross-validation
)
# Extract coefficients for the best lambda (lambda.1se for simplicity)
coefs <- coef(cv, s = cv$lambda.1se)
# Print the best lambda
best_lambda <- cv$lambda.min
print(paste("Best lambda:", best_lambda))
```

```
[1] "Best lambda: 0.602088555655706"
```

```
# Extract non-zero coefficient variables (important features)
fre.variables <- rownames(coefs)[coefs[, 1] != 0]
fre.variables <- fre.variables[fre.variables != "(Intercept)"]  # Exclude intercept
print("Selected features:")
```

```
[1] "Selected features:"
```

```
print(fre.variables)
```

```
[1] "BMI.CategoryOverweight" "BloodPressure_Upper"    "BloodPressure_Lower"   
```

```
#------------------------------手選變數

logist_self<-train(
  Sleep.Disorder ~ BloodPressure_Upper + Stress.Level + Sleep.Duration +  BMI.Category,
  data = data,
  method = "glm",  # Specify "multinom" for multinomial logistic regression
  family = "binomial",       # Specify binary outcome
  trControl = train_control,
)
logist_self$results
```

```
  parameter  Accuracy     Kappa AccuracySD    KappaSD
1      none 0.9437372 0.8835882 0.02987052 0.06211252
```

```
print(logist_self)
```

```
Generalized Linear Model

374 samples
  4 predictor
  2 classes: '0', '1'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 337, 337, 337, 337, 336, 336, ...
Resampling results:

  Accuracy   Kappa
  0.9437372  0.8835882
```

```
# view final model 最終決定的模型，以及模型估計係數值
logist_self$finalModel
```

```

Call:  NULL

Coefficients:
           (Intercept)     BloodPressure_Upper           Stress.Level4
             -38.71154                 0.25730                 2.07689
         Stress.Level5           Stress.Level6           Stress.Level7
               0.09856                 0.40956                 4.29436
         Stress.Level8          Sleep.Duration  BMI.CategoryOverweight
               1.66040                 0.40486                 1.60522

Degrees of Freedom: 373 Total (i.e. Null);  365 Residual
Null Deviance:      507.5
Residual Deviance: 172.3    AIC: 190.3
```

```
#view predictions for each fold，每一折（fold）/子集（subset）資料的預測誤差
logist_self$resample
```

```
    Accuracy     Kappa Resample
1  0.8918919 0.7757576   Fold01
2  0.9459459 0.8878788   Fold02
3  0.8918919 0.7757576   Fold03
4  0.9459459 0.8878788   Fold04
5  0.9736842 0.9455587   Fold05
6  0.9473684 0.8901734   Fold06
```

```
7  0.9473684 0.8920455   Fold07
8  0.9473684 0.8920455   Fold08
9  0.9736842 0.9464789   Fold09
10 0.9722222 0.9423077   Fold10
```

cross validation(repeated k-fold)

```
train.rkfold <- trainControl(method = "repeatedcv", number = 5, repeats = 3)

logist1<-train(
  Sleep.Disorder ~ .,
  data = data,
  method = "glm",  # Specify "multinom" for multinomial logistic regression
  family = "binomial",        # Specify binary outcome
  trControl = train.rkfold,
)
```

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

```
print(logist1)
```

```
Generalized Linear Model

374 samples
 12 predictor
  2 classes: '0', '1'

No pre-processing
Resampling: Cross-Validated (5 fold, repeated 3 times)
Summary of sample sizes: 299, 299, 299, 300, 299, 299, ...
Resampling results:

  Accuracy   Kappa
  0.9304985  0.8570002
```

```
logist1$results
```

```
  parameter  Accuracy     Kappa AccuracySD    KappaSD
1      none 0.9304985 0.8570002 0.02945747 0.05957323
```