Depression Draft 1

Christy Hui

2024-11-30

```r
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.4.1

## Warning: package 'ggplot2' was built under R version 4.4.1

## Warning: package 'tidyr' was built under R version 4.4.1

## Warning: package 'readr' was built under R version 4.4.1

## Warning: package 'purrr' was built under R version 4.4.1

## Warning: package 'stringr' was built under R version 4.4.1

## Warning: package 'forcats' was built under R version 4.4.1

## Warning: package 'lubridate' was built under R version 4.4.1

## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 4.4.2

##
## Attaching package: 'cowplot'
##
## The following object is masked from 'package:lubridate':
##
##     stamp
```

```r
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.4.1

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

```
##
##     lift
```

```r
library(ROCR)
```

```
## Warning: package 'ROCR' was built under R version 4.4.2
```

```r
library(sjPlot)
```

```
## Warning: package 'sjPlot' was built under R version 4.4.2
```

```
##
## Attaching package: 'sjPlot'
##
## The following objects are masked from 'package:cowplot':
##
##     plot_grid, save_plot
```

```r
library(visdat)
```

```
## Warning: package 'visdat' was built under R version 4.4.2
```

```r
# read data
depression = read.csv("final_depression_dataset_1.csv")
```

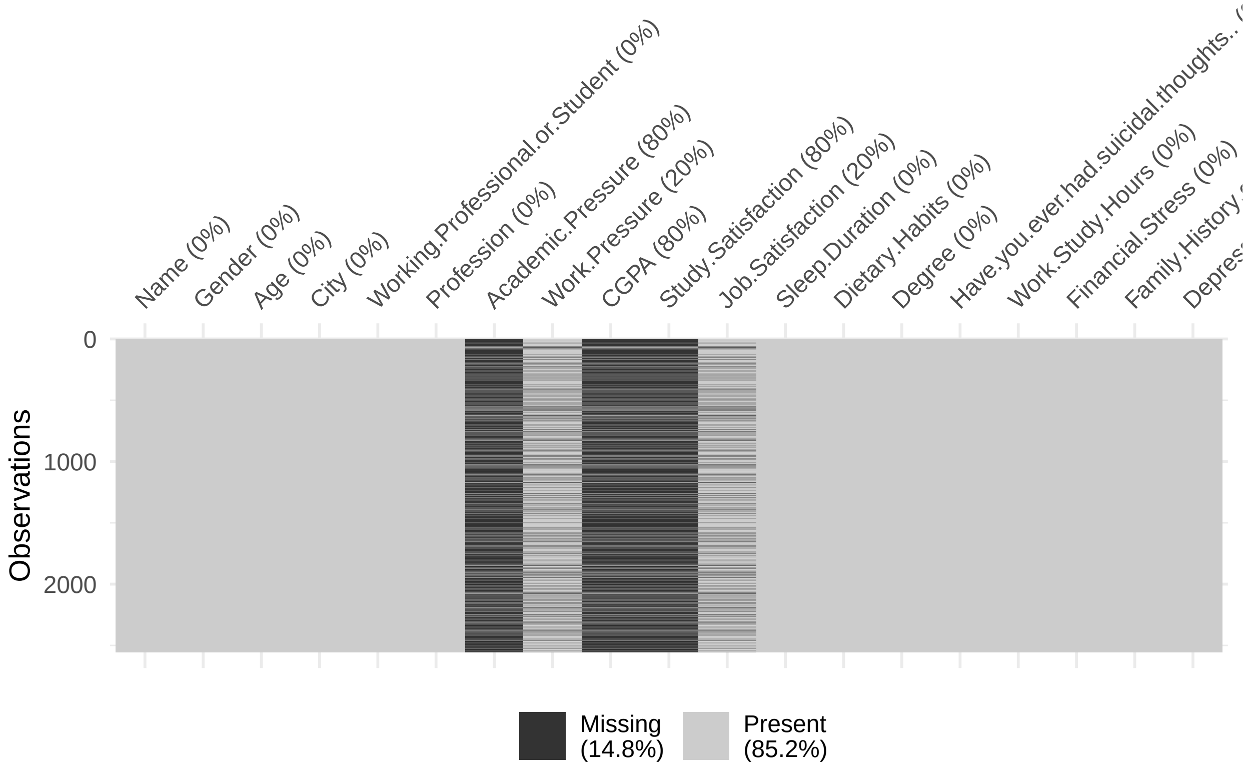
```r
# find the dimension of depression
dim(depression)
```

```
## [1] 2556   19
```

```r
# find if there exist duplicates
sum(duplicated(depression))
```

```
## [1] 0
```

```r
vis_miss(depression)
```



```r
# find number of NAs for each column
sapply(depression, function(x) {sum(is.na(x))})
```

```
##                                  Name                                Gender
##                                     0                                     0
##                                   Age                                  City
##                                     0                                     0
##       Working.Professional.or.Student                            Profession
##                                     0                                     0
##                     Academic.Pressure                         Work.Pressure
##                                  2054                                   502
##                                  CGPA                    Study.Satisfaction
##                                  2054                                  2054
##                      Job.Satisfaction                        Sleep.Duration
##                                   502                                     0
##                        Dietary.Habits                                Degree
##                                     0                                     0
## Have.you.ever.had.suicidal.thoughts..                      Work.Study.Hours
##                                     0                                     0
##                      Financial.Stress      Family.History.of.Mental.Illness
##                                     0                                     0
##                            Depression
##                                     0
```

```r
# combine pressure columns into one
helper1 = ifelse(is.na(depression$Academic.Pressure), 0, depression$Academic.Pressure)
```

```r
helper2 = ifelse(is.na(depression$Work.Pressure), 0, depression$Work.Pressure)

depression$Pressure = helper1 + helper2
```

```r
# combine satisfaction into one column
helper3 = ifelse(is.na(depression$Study.Satisfaction), 0, depression$Study.Satisfaction)

helper4 = ifelse(is.na(depression$Job.Satisfaction), 0, depression$Job.Satisfaction)

depression$Satisfaction = helper3 + helper4
```

```r
# delete columns with NAs
depression = depression[, -c(7:11)]
sapply(depression, function(x) {sum(is.na(x))})
```

```
##                                  Name                                Gender
##                                     0                                     0
##                                   Age                                  City
##                                     0                                     0
##       Working.Professional.or.Student                            Profession
##                                     0                                     0
##                        Sleep.Duration                        Dietary.Habits
##                                     0                                     0
##                                Degree Have.you.ever.had.suicidal.thoughts..
##                                     0                                     0
##                      Work.Study.Hours                      Financial.Stress
##                                     0                                     0
##      Family.History.of.Mental.Illness                            Depression
##                                     0                                     0
##                              Pressure                          Satisfaction
##                                     0                                     0
```

```r
# due to a large amount of varied answers for "City" and "Profession," we delete the variables
# we also delete name because we don't care about that variable
unique(depression$City)
```

```
##  [1] "Ghaziabad"     "Kalyan"        "Bhopal"        "Thane"
##  [5] "Indore"        "Pune"          "Bangalore"     "Hyderabad"
##  [9] "Srinagar"      "Nashik"        "Kolkata"       "Ahmedabad"
## [13] "Varanasi"      "Chennai"       "Jaipur"        "Surat"
## [17] "Vasai-Virar"   "Rajkot"        "Patna"         "Mumbai"
## [21] "Vadodara"      "Lucknow"       "Faridabad"     "Meerut"
## [25] "Kanpur"        "Visakhapatnam" "Ludhiana"      "Nagpur"
## [29] "Delhi"         "Agra"
```

```r
unique(depression$Profession)
```

```
##  [1] "Teacher"                "Financial Analyst"      "UX/UI Designer"
##  [4] "Civil Engineer"         "Accountant"             "Lawyer"
##  [7] "Content Writer"         ""                       "Pilot"
## [10] "Customer Support"       "Judge"                  "Architect"
## [13] "HR Manager"             "Digital Marketer"       "Sales Executive"
## [16] "Business Analyst"       "Mechanical Engineer"    "Consultant"
## [19] "Data Scientist"         "Pharmacist"             "Software Engineer"
## [22] "Travel Consultant"      "Manager"                "Entrepreneur"
## [25] "Doctor"                 "Researcher"             "Plumber"
## [28] "Finanancial Analyst"    "Marketing Manager"      "Educational Consultant"
## [31] "Chemist"                "Research Analyst"       "Chef"
## [34] "Electrician"            "Graphic Designer"       "Investment Banker"
```

```
depression = subset(depression, select = -c(Name, City, Profession))
```

```
# degree has many varied answers as well; however, they can be recoded into three main categories: high
unique(depression$Degree)
```

```
##  [1] "MA"       "B.Com"    "M.Com"    "MD"       "BE"       "MCA"
##  [7] "BA"       "LLM"      "BCA"      "Class 12" "B.Ed"     "M.Tech"
## [13] "LLB"      "B.Arch"   "ME"       "MBA"      "M.Pharm"  "MBBS"
## [19] "PhD"      "BSc"      "MSc"      "MHM"      "BBA"      "BHM"
## [25] "B.Tech"   "M.Ed"     "B.Pharm"
```

```
# recode degree into three categories
depression$Degree = case_when(depression$Degree == "Class 12" ~ "High School Equivalent",
                              grepl("^[BL]", depression$Degree) ~ "Bachelors Degree",
                              grepl("^[MP]", depression$Degree) ~ "Post-Graduate Degree")

table(depression$Degree)
```

```
## 
##       Bachelors Degree High School Equivalent   Post-Graduate Degree 
##                   1193                    275                   1088
```

```
# find type of each variable so we can change each type
sapply(depression, function(x) {class(x)})
```

```
##                                Gender                                   Age 
##                           "character"                             "integer" 
##       Working.Professional.or.Student                        Sleep.Duration 
##                           "character"                           "character" 
##                        Dietary.Habits                                Degree 
##                           "character"                           "character" 
## Have.you.ever.had.suicidal.thoughts..                      Work.Study.Hours 
##                           "character"                             "integer" 
##                      Financial.Stress      Family.History.of.Mental.Illness 
##                             "integer"                           "character" 
##                            Depression                              Pressure 
##                           "character"                             "numeric" 
##                          Satisfaction 
##                             "numeric"
```

```
# change each categorical into a factor, changing the base/ordering them if needed
depression$Gender = as.factor(depression$Gender)
depression$Working.Professional.or.Student = as.factor(depression$Working.Professional.or.Student)
depression$Sleep.Duration = factor(depression$Sleep.Duration, levels = c("Less than 5 hours", "5-6 hour
depression$Dietary.Habits = factor(depression$Dietary.Habits, levels = c("Unhealthy", "Moderate", "Heal
depression$Degree = factor(depression$Degree, levels = c("High School Equivalent", "Bachelors Degree",
depression$Have.you.ever.had.suicidal.thoughts.. = as.factor(depression$Have.you.ever.had.suicidal.thou
depression$Financial.Stress = factor(depression$Financial.Stress, levels = c(1, 2, 3, 4, 5), ordered = 
depression$Family.History.of.Mental.Illness = as.factor(depression$Family.History.of.Mental.Illness)
depression$Depression = as.factor(depression$Depression)
depression$Pressure = factor(depression$Pressure, levels = c(1, 2, 3, 4, 5), ordered = TRUE)
depression$Satisfaction = factor(depression$Satisfaction, levels = c(1, 2, 3, 4, 5), ordered = TRUE)
```

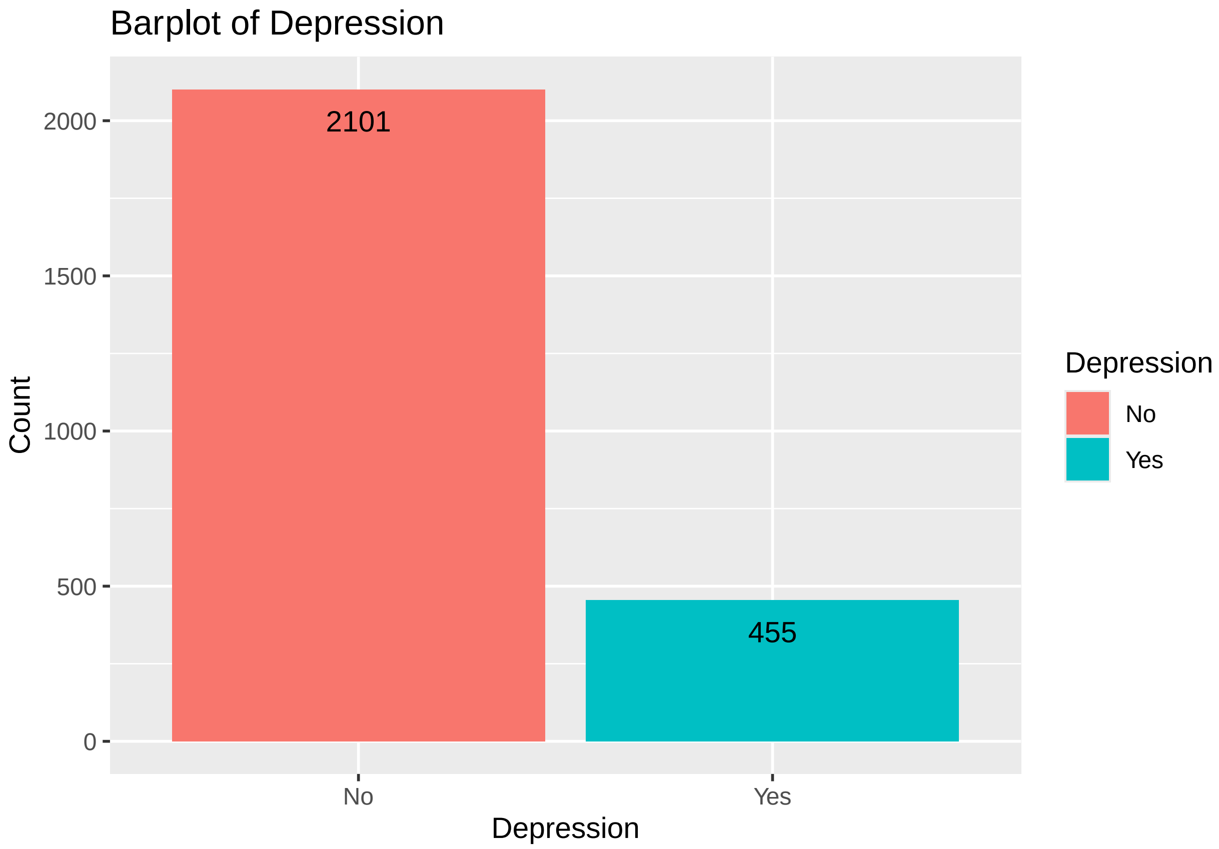
```r
# find if any variables are unbalanced
depressionFactored = select(depression, where(is.factor))
sapply(depressionFactored, table)
```

```
## $Gender
## 
## Female   Male 
##   1223   1333 
## 
## $Working.Professional.or.Student
## 
##               Student Working Professional 
##                   502                 2054 
## 
## $Sleep.Duration
## 
## Less than 5 hours         5-6 hours         7-8 hours More than 8 hours 
##               648               628               658               622 
## 
## $Dietary.Habits
## 
## Unhealthy  Moderate   Healthy 
##       882       832       842 
## 
## $Degree
## 
## High School Equivalent       Bachelors Degree   Post-Graduate Degree 
##                    275                   1193                   1088 
## 
## $Have.you.ever.had.suicidal.thoughts..
## 
##   No  Yes 
## 1307 1249 
## 
## $Financial.Stress
## 
##   1   2   3   4   5 
## 517 549 488 501 501 
## 
## $Family.History.of.Mental.Illness
## 
##   No  Yes 
## 1311 1245 
## 
## $Depression
## 
##   No  Yes 
## 2101  455 
## 
## $Pressure
## 
##   1   2   3   4   5 
## 500 501 529 504 522 
## 
```

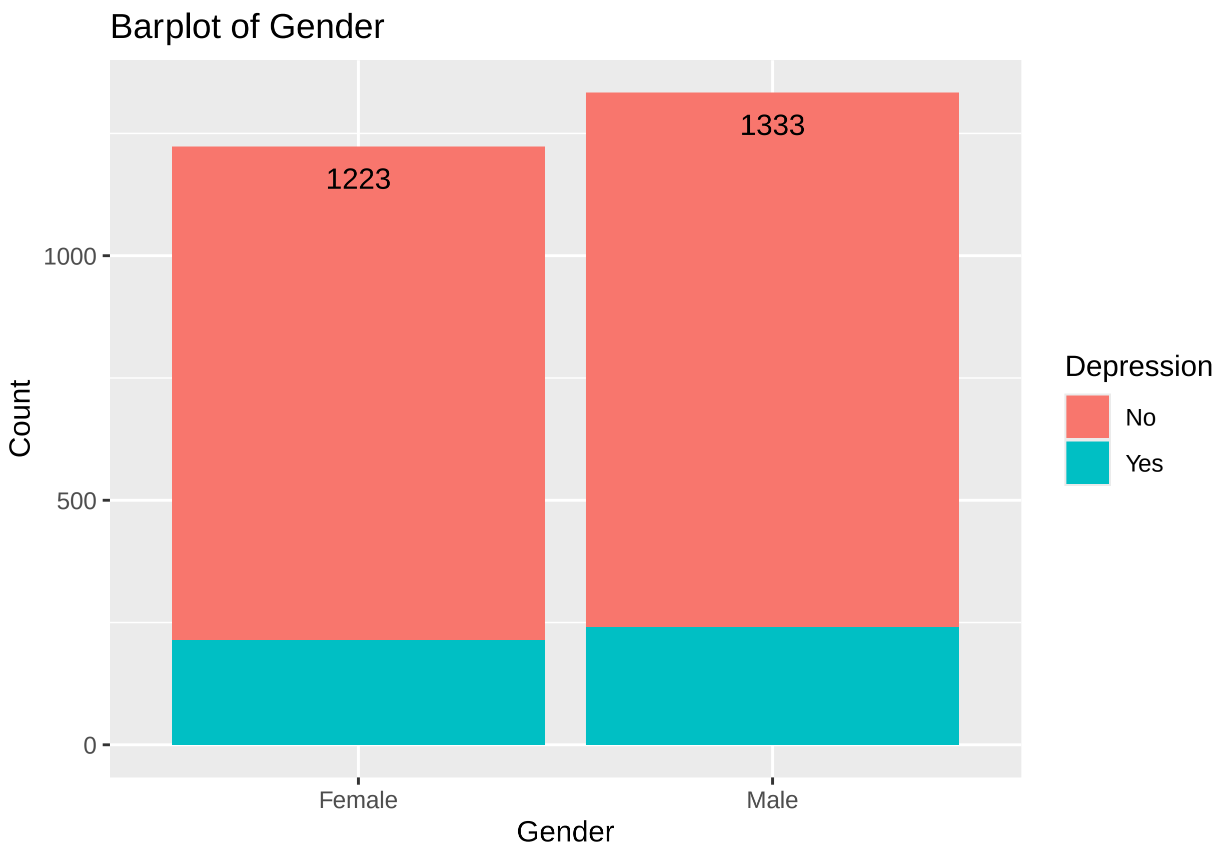
```
## $Satisfaction
##
##   1   2   3   4   5
## 482 531 507 508 528
```

```r
# plot depression count
ggplot(depression, aes(x = Depression)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Depression") +
  ylab("Count") +
  ggtitle("Barplot of Depression") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```

```
## Warning: The dot-dot notation (`..count..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(count)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



```r
# plot gender
ggplot(depression, aes(x = Gender)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Gender") +
  ylab("Count") +
  ggtitle("Barplot of Gender") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



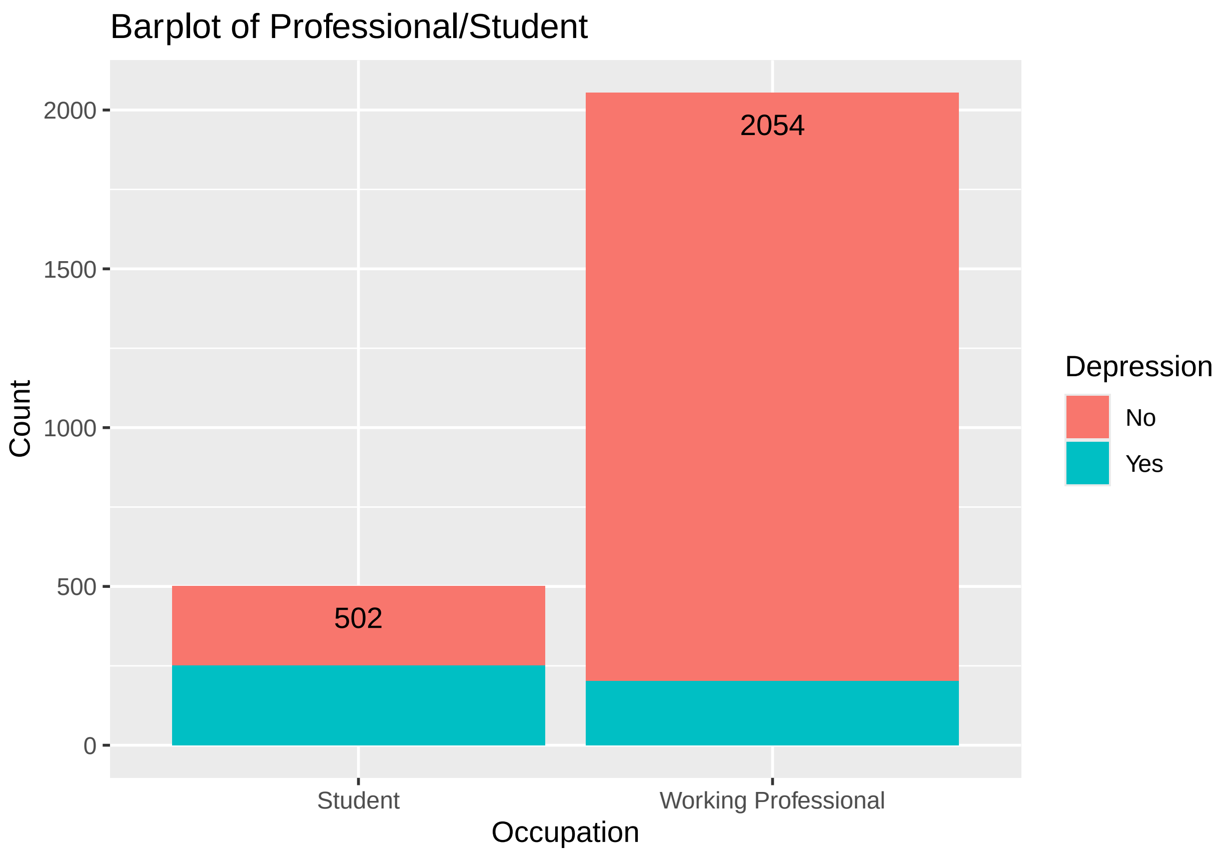
```
table(depression$Depression, depression$Gender)
```

```
##
##       Female Male
##   No    1009 1092
##   Yes    214  241
```

```
prop.table(table(depression$Depression, depression$Gender), margin = 1)
```

```
##
##          Female      Male
##   No  0.4802475 0.5197525
##   Yes 0.4703297 0.5296703
```

```
# plot whether or not person is a working professional or student
ggplot(depression, aes(x = Working.Professional.or.Student)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Occupation") +
  ylab("Count") +
  ggtitle("Barplot of Professional/Student") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



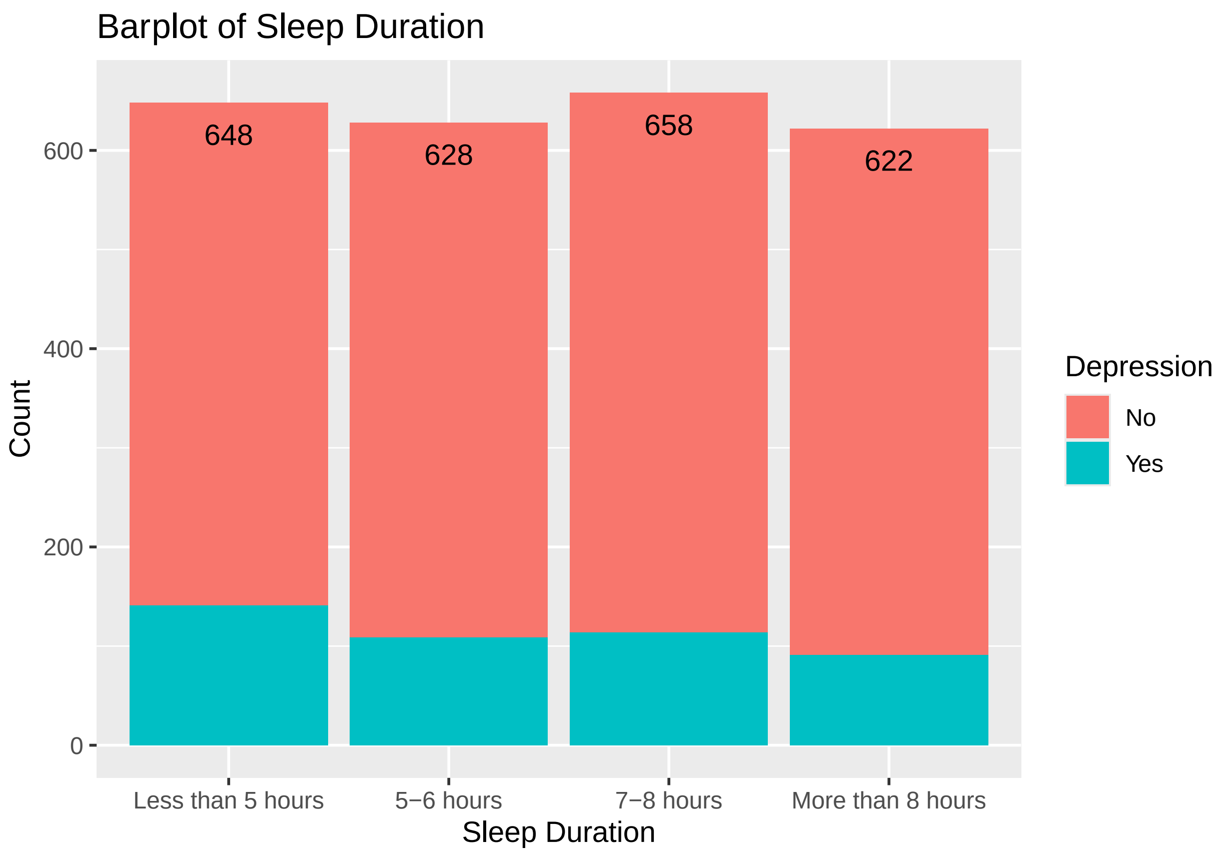
```r
table(depression$Depression, depression$Working.Professional.or.Student)
```

```
##
##       Student Working Professional
##   No      250                 1851
##   Yes     252                  203
```

```r
prop.table(table(depression$Depression, depression$Working.Professional.or.Student), margin = 1)
```

```
##
##         Student Working Professional
##   No  0.1189910            0.8810090
##   Yes 0.5538462            0.4461538
```

```r
# plot sleep duration habits
ggplot(depression, aes(x = Sleep.Duration)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Sleep Duration") +
  ylab("Count") +
  ggtitle("Barplot of Sleep Duration") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



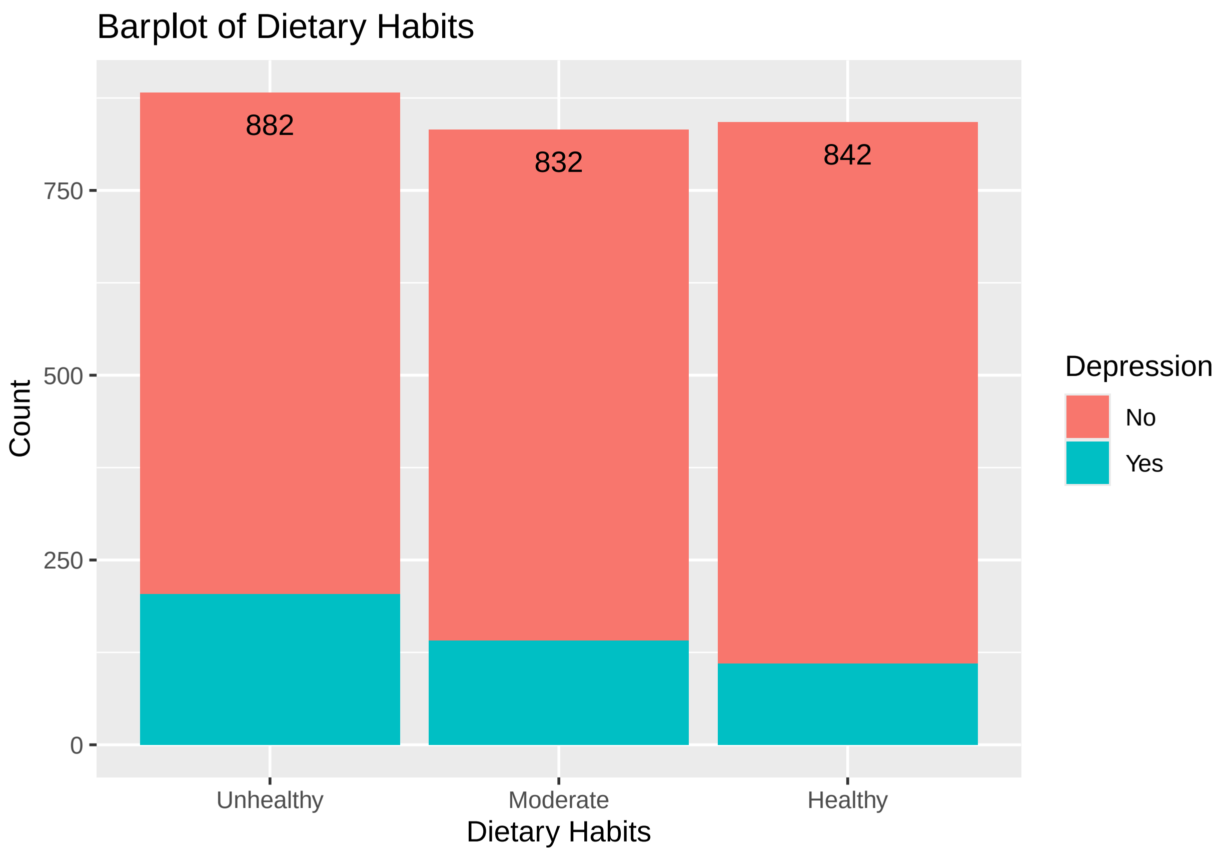
```
table(depression$Depression, depression$Sleep.Duration)
```

```
##
##       Less than 5 hours 5-6 hours 7-8 hours More than 8 hours
##   No                507       519       544               531
##   Yes               141       109       114                91
```

```
prop.table(table(depression$Depression, depression$Sleep.Duration), margin = 1)
```

```
##
##       Less than 5 hours 5-6 hours 7-8 hours More than 8 hours
##   No          0.2413137 0.2470252 0.2589243         0.2527368
##   Yes         0.3098901 0.2395604 0.2505495         0.2000000
```

```
# plot dietary habits
ggplot(depression, aes(x = Dietary.Habits)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Dietary Habits") +
  ylab("Count") +
  ggtitle("Barplot of Dietary Habits") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



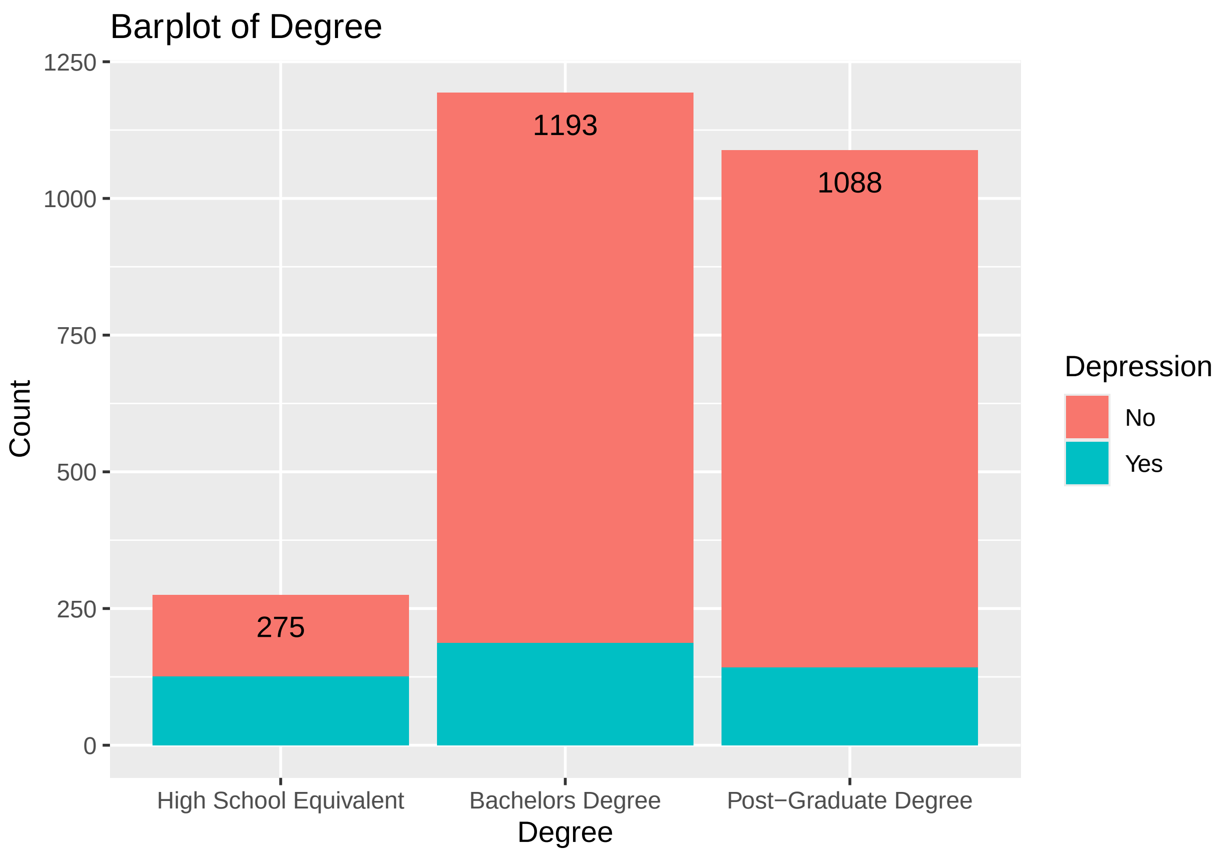
```
table(depression$Depression, depression$Dietary.Habits)
```

```
##
##       Unhealthy Moderate Healthy
##   No        678      691     732
##   Yes       204      141     110
```

```
prop.table(table(depression$Depression, depression$Dietary.Habits), margin = 1)
```

```
##
##       Unhealthy  Moderate   Healthy
##   No  0.3227035 0.3288910 0.3484055
##   Yes 0.4483516 0.3098901 0.2417582
```

```
# plot degree count
ggplot(depression, aes(x = Degree)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Degree") +
  ylab("Count") +
  ggtitle("Barplot of Degree") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



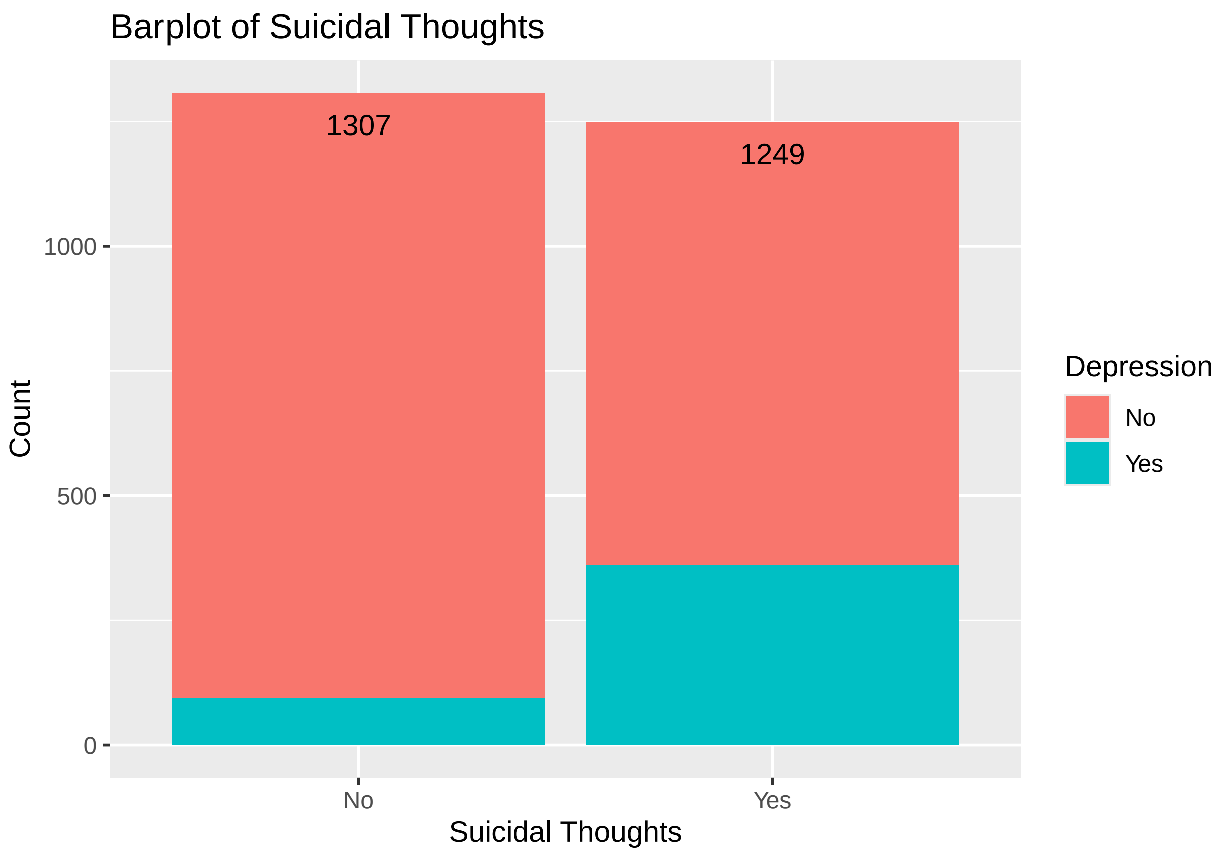
```
table(depression$Depression, depression$Degree)
```

```
##
##       High School Equivalent Bachelors Degree Post-Graduate Degree
##   No                     149             1006                  946
##   Yes                    126              187                  142
```

```
prop.table(table(depression$Depression, depression$Degree), margin = 1)
```

```
##
##       High School Equivalent Bachelors Degree Post-Graduate Degree
##   No              0.07091861       0.47881961           0.45026178
##   Yes             0.27692308       0.41098901           0.31208791
```

```
# plot degree count
ggplot(depression, aes(x = Have.you.ever.had.suicidal.thoughts..)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Suicidal Thoughts") +
  ylab("Count") +
  ggtitle("Barplot of Suicidal Thoughts") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



```
table(depression$Depression, depression$Have.you.ever.had.suicidal.thoughts..)
```

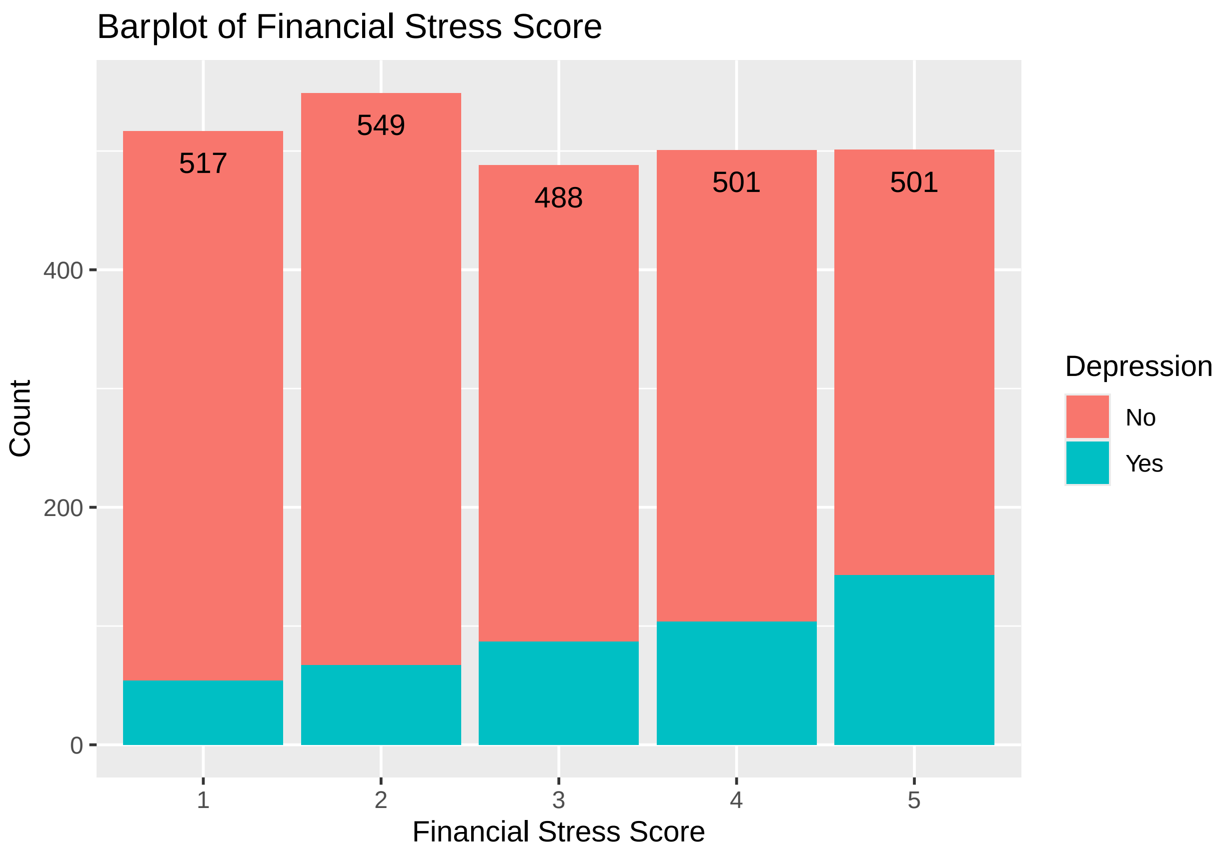
```
##
##          No  Yes
##   No   1212  889
##   Yes    95  360
```

```
prop.table(table(depression$Depression, depression$Have.you.ever.had.suicidal.thoughts..), margin = 1)
```

```
##
##               No       Yes
##   No   0.5768682 0.4231318
##   Yes  0.2087912 0.7912088
```

```
# delete suicidal thoughts variable
depression = subset(depression, select = -c(Have.you.ever.had.suicidal.thoughts..))
```

```
# plot financial stress count
ggplot(depression, aes(x = Financial.Stress)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Financial Stress Score") +
  ylab("Count") +
  ggtitle("Barplot of Financial Stress Score") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



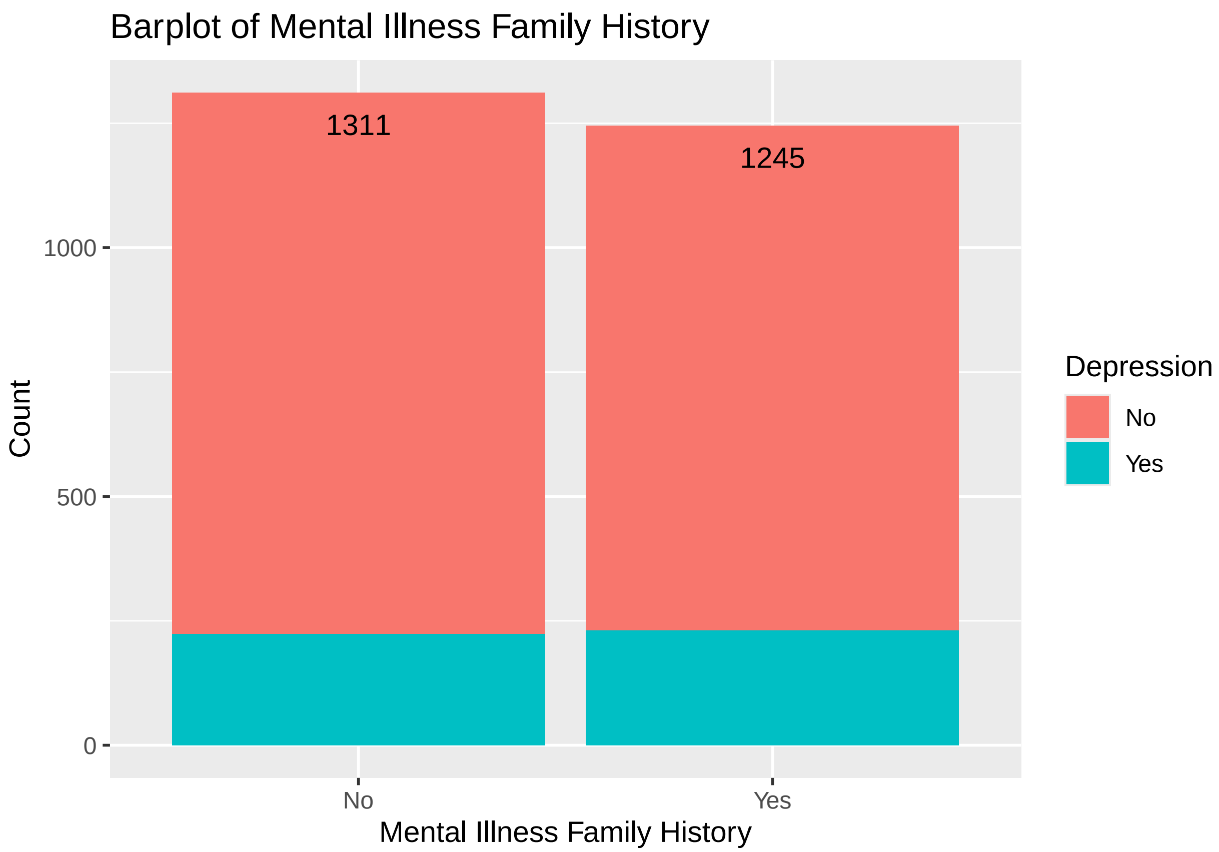
```
table(depression$Depression, depression$Financial.Stress)
```

```
##
##         1   2   3   4   5
##   No  463 482 401 397 358
##   Yes  54  67  87 104 143
```

```
prop.table(table(depression$Depression, depression$Financial.Stress), margin = 1)
```

```
##
##               1         2         3         4         5
##   No  0.2203713 0.2294146 0.1908615 0.1889576 0.1703950
##   Yes 0.1186813 0.1472527 0.1912088 0.2285714 0.3142857
```

```
# plot family history of mental illness count
ggplot(depression, aes(x = Family.History.of.Mental.Illness)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Mental Illness Family History") +
  ylab("Count") +
  ggtitle("Barplot of Mental Illness Family History") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



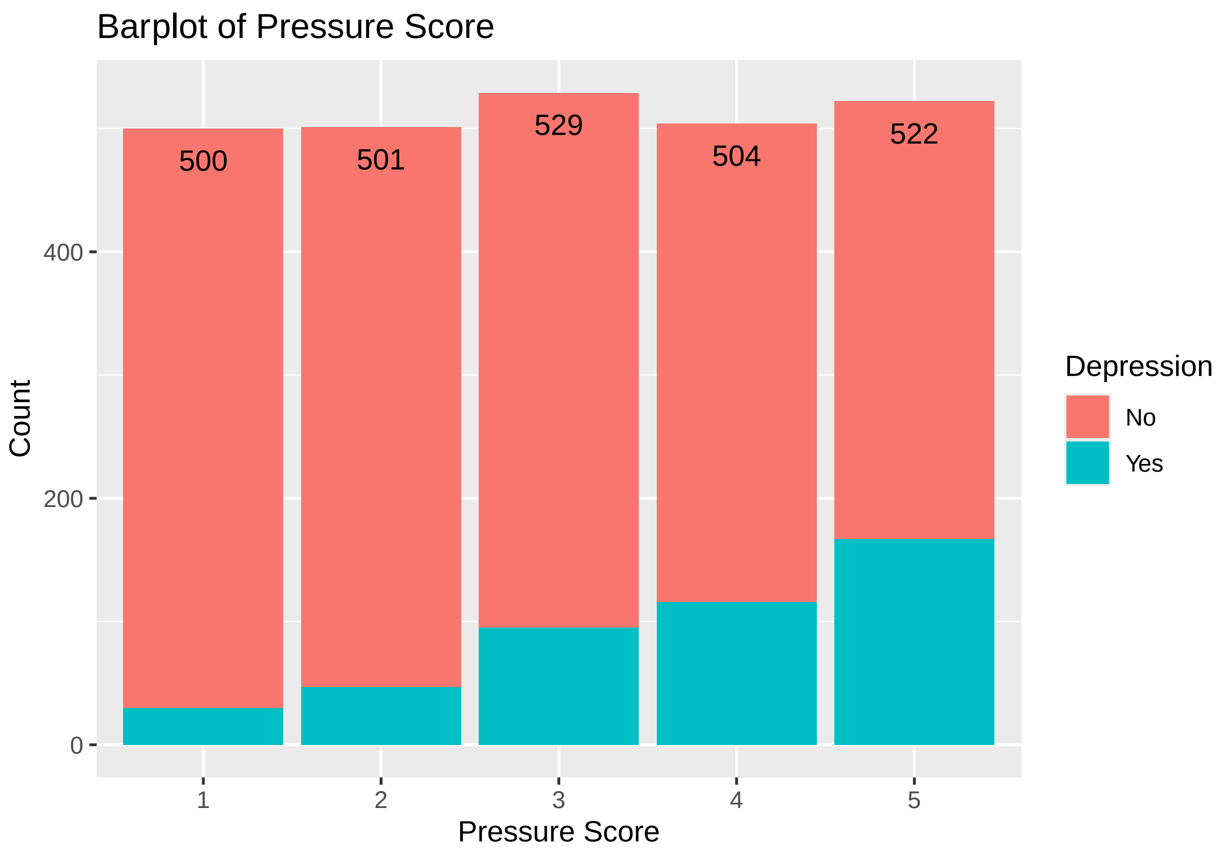
```
table(depression$Depression, depression$Family.History.of.Mental.Illness)
```

```
##
##         No  Yes
##   No  1087 1014
##   Yes  224  231
```

```
prop.table(table(depression$Depression, depression$Family.History.of.Mental.Illness), margin = 1)
```

```
##
##              No       Yes
##   No  0.5173727 0.4826273
##   Yes 0.4923077 0.5076923
```

```
# plot financial stress count
ggplot(depression, aes(x = Pressure)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Pressure Score") +
  ylab("Count") +
  ggtitle("Barplot of Pressure Score") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



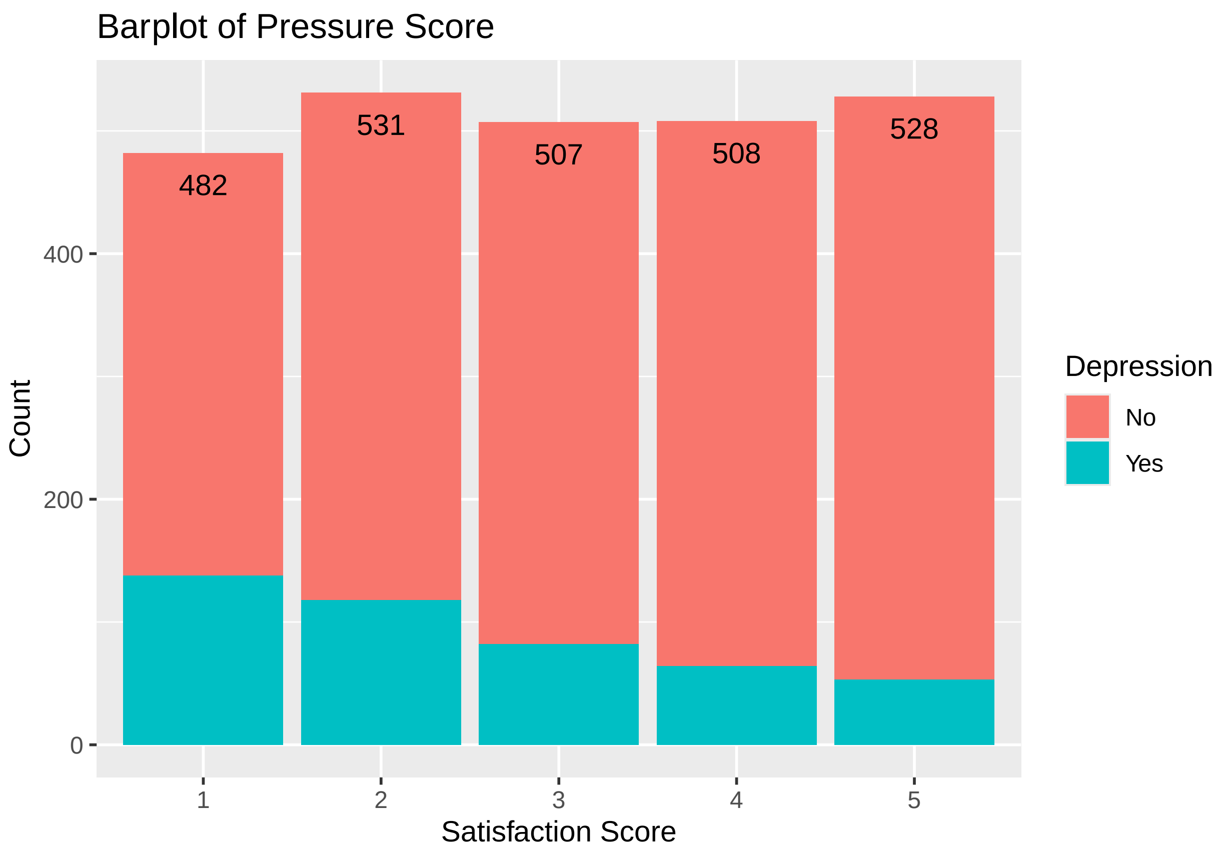
```
table(depression$Depression, depression$Pressure)
```

```
##
##         1   2   3   4   5
##   No  470 454 434 388 355
##   Yes  30  47  95 116 167
```

```
prop.table(table(depression$Depression, depression$Pressure), margin = 1)
```

```
##
##                1          2          3          4          5
##   No  0.22370300 0.21608758 0.20656830 0.18467396 0.16896716
##   Yes 0.06593407 0.10329670 0.20879121 0.25494505 0.36703297
```

```
ggplot(depression, aes(x = Satisfaction)) +
  geom_bar(aes(fill = Depression)) +
  xlab("Satisfaction Score") +
  ylab("Count") +
  ggtitle("Barplot of Pressure Score") +
  geom_text(aes(label = ..count..), stat = "count", vjust = 2)
```



```r
table(depression$Depression, depression$Satisfaction)
```

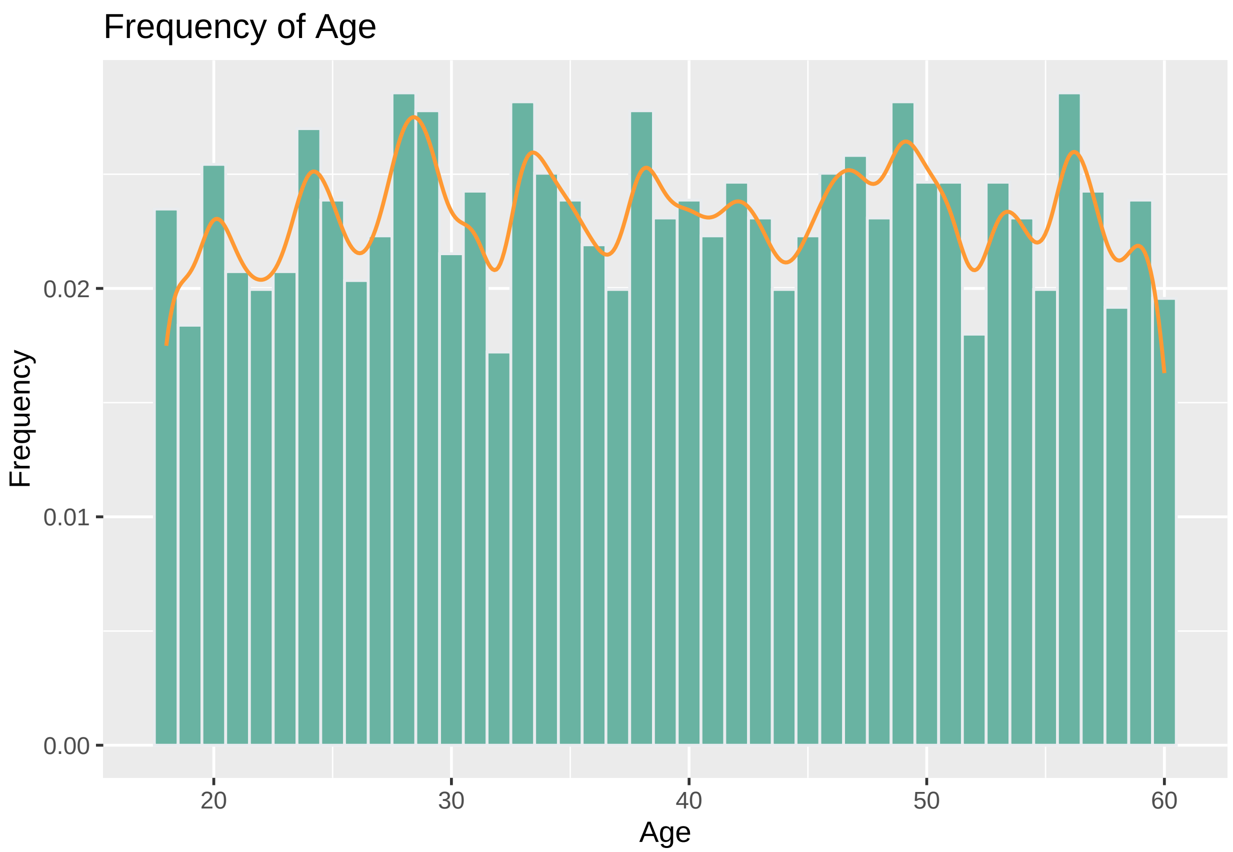
```
##      
##         1   2   3   4   5
##   No  344 413 425 444 475
##   Yes 138 118  82  64  53
```

```r
prop.table(table(depression$Depression, depression$Satisfaction), margin = 1)
```

```
##      
##               1         2         3         4         5
##   No  0.1637316 0.1965731 0.2022846 0.2113279 0.2260828
##   Yes 0.3032967 0.2593407 0.1802198 0.1406593 0.1164835
```

```r
# create specific data frames to separate those with and without risk of depression
depressionYes = depression[depression$Depression == "Yes", ]
depressionNo = depression[depression$Depression == "No", ]
```

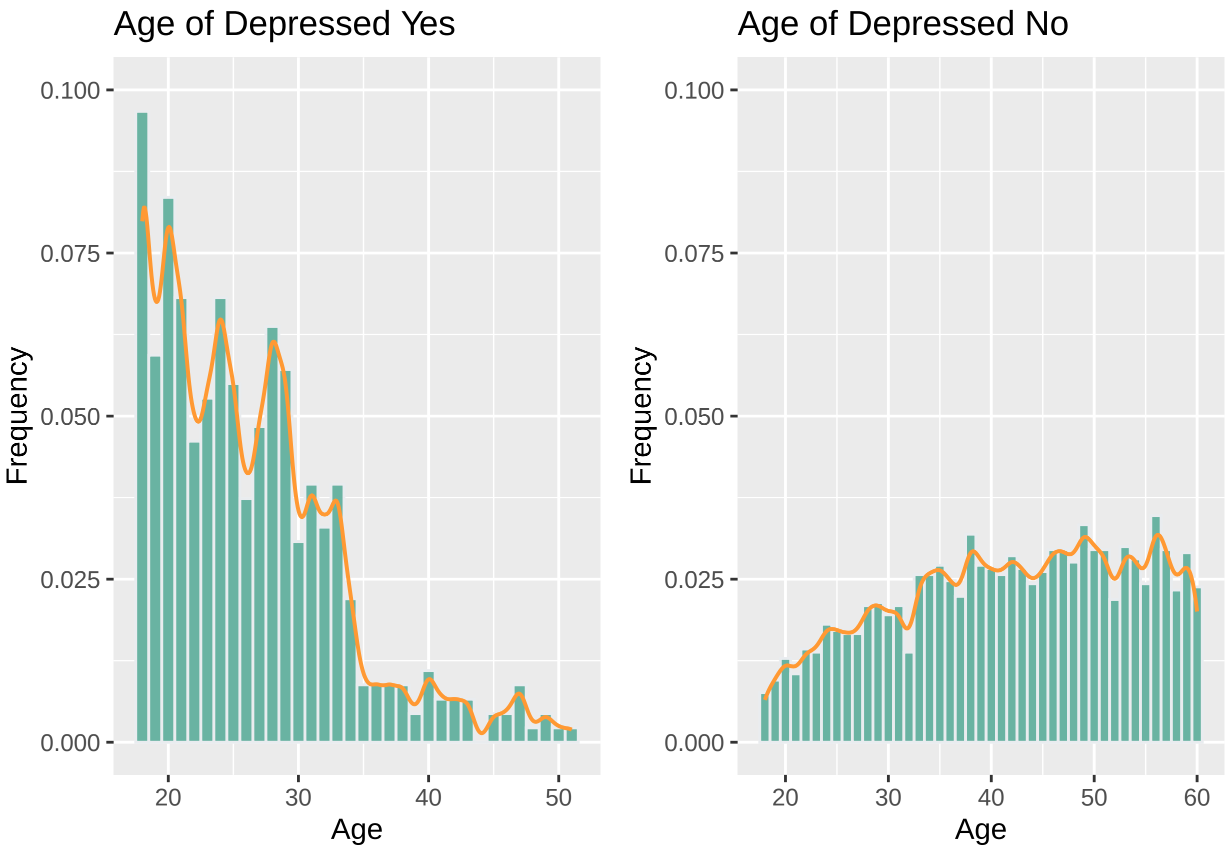
```r
ggplot(depression, aes(x = Age, y = after_stat(density))) +
  geom_histogram(binwidth = 1, fill="#69B3A2", color = "#E9ECEF") +
  geom_density(color = "#FF9933", linewidth = 0.7, adjust = 0.3) +
  ggtitle("Frequency of Age") +
  ylab("Frequency")
```



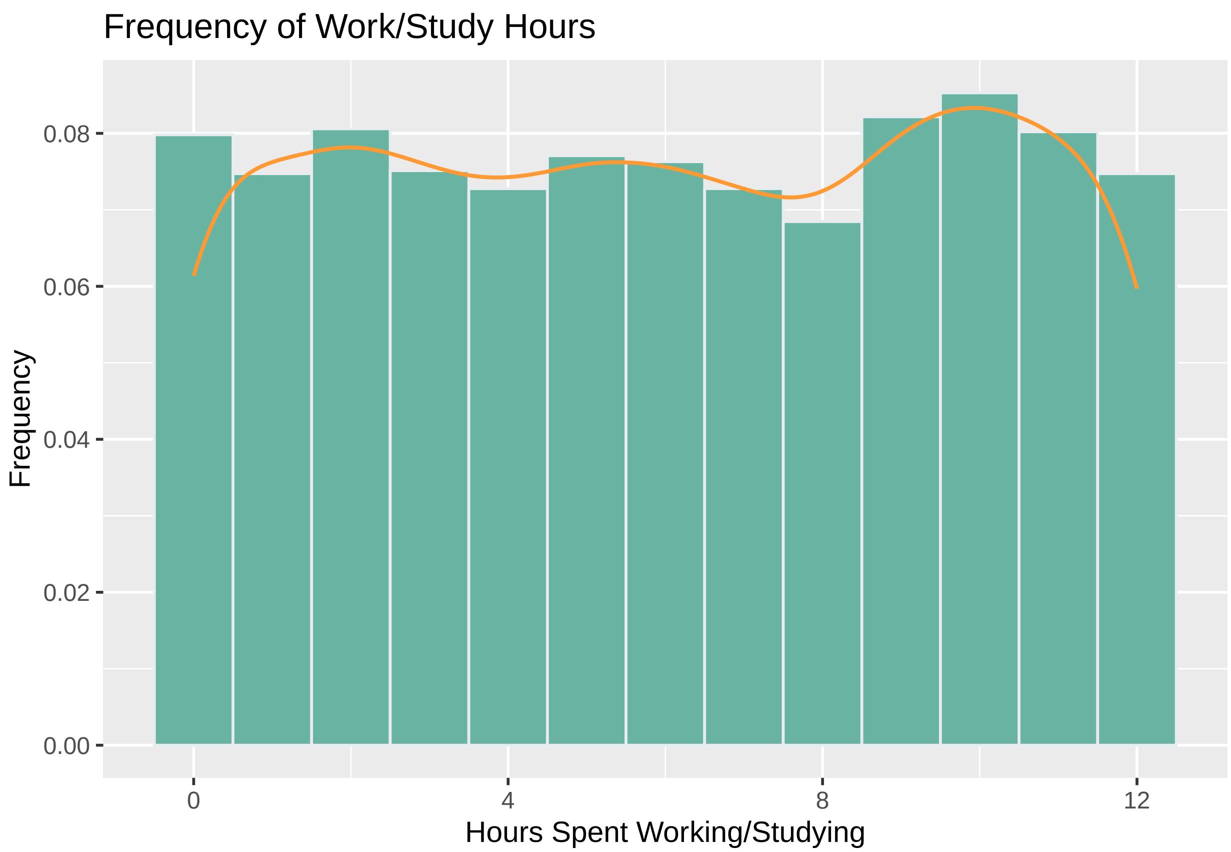
```
p1 = ggplot(depressionYes, aes(x = Age, y = after_stat(density))) +
  geom_histogram(binwidth = 1, fill="#69B3A2", color = "#E9ECEF") +
  geom_density(color = "#FF9933", linewidth = 0.7, adjust = 0.3) +
  ggtitle("Age of Depressed Yes") +
  ylab("Frequency") +
  ylim(0, 0.10)

p2 = ggplot(depressionNo, aes(x = Age, y = after_stat(density))) +
  geom_histogram(binwidth = 1, fill="#69B3A2", color = "#E9ECEF") +
  geom_density(color = "#FF9933", linewidth = 0.7, adjust = 0.3) +
  ggtitle("Age of Depressed No") +
  ylab("Frequency") +
  ylim(0, 0.10)
```

```
cowplot::plot_grid(p1, p2)
```



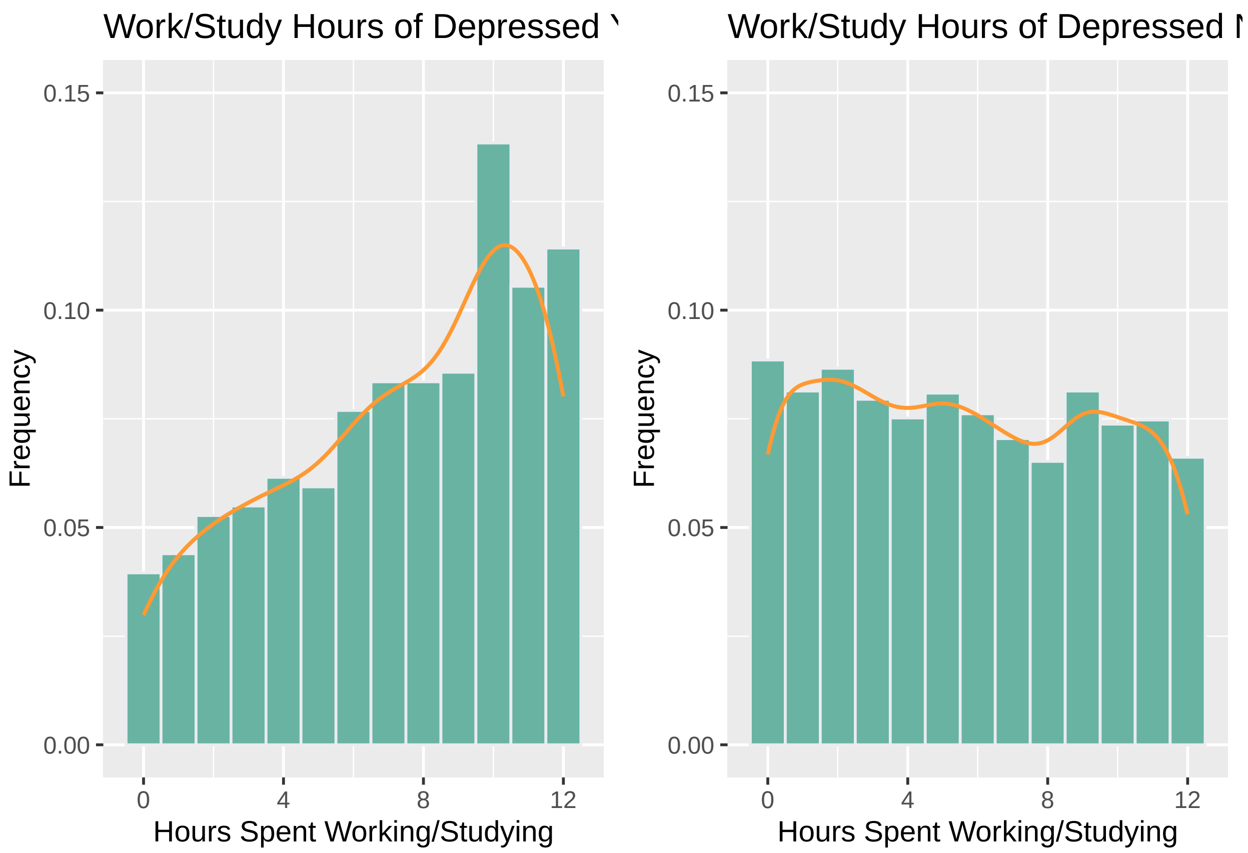
```
ggplot(depression, aes(x = Work.Study.Hours, y = after_stat(density))) +
  geom_histogram(binwidth = 1, fill="#69B3A2", color = "#E9ECEF") +
  geom_density(color = "#FF9933", linewidth = 0.7, adjust = 1) +
  ggtitle("Frequency of Work/Study Hours") +
  xlab("Hours Spent Working/Studying") +
  ylab("Frequency")
```



```
p3 = ggplot(depressionYes, aes(x = Work.Study.Hours, y = after_stat(density))) +
  geom_histogram(binwidth = 1, fill="#69B3A2", color = "#E9ECEF") +
  geom_density(color = "#FF9933", linewidth = 0.7, adjust = 1) +
  ggtitle("Work/Study Hours of Depressed Yes") +
  xlab("Hours Spent Working/Studying") +
  ylab("Frequency") +
  ylim(0, 0.15)

p4 = ggplot(depressionNo, aes(x = Work.Study.Hours, y = after_stat(density))) +
  geom_histogram(binwidth = 1, fill="#69B3A2", color = "#E9ECEF") +
  geom_density(color = "#FF9933", linewidth = 0.7, adjust = 1) +
  ggtitle("Work/Study Hours of Depressed No") +
  xlab("Hours Spent Working/Studying") +
  ylab("Frequency") +
  ylim(0, 0.15)
```

```
cowplot::plot_grid(p3, p4, ncol = 2)
```



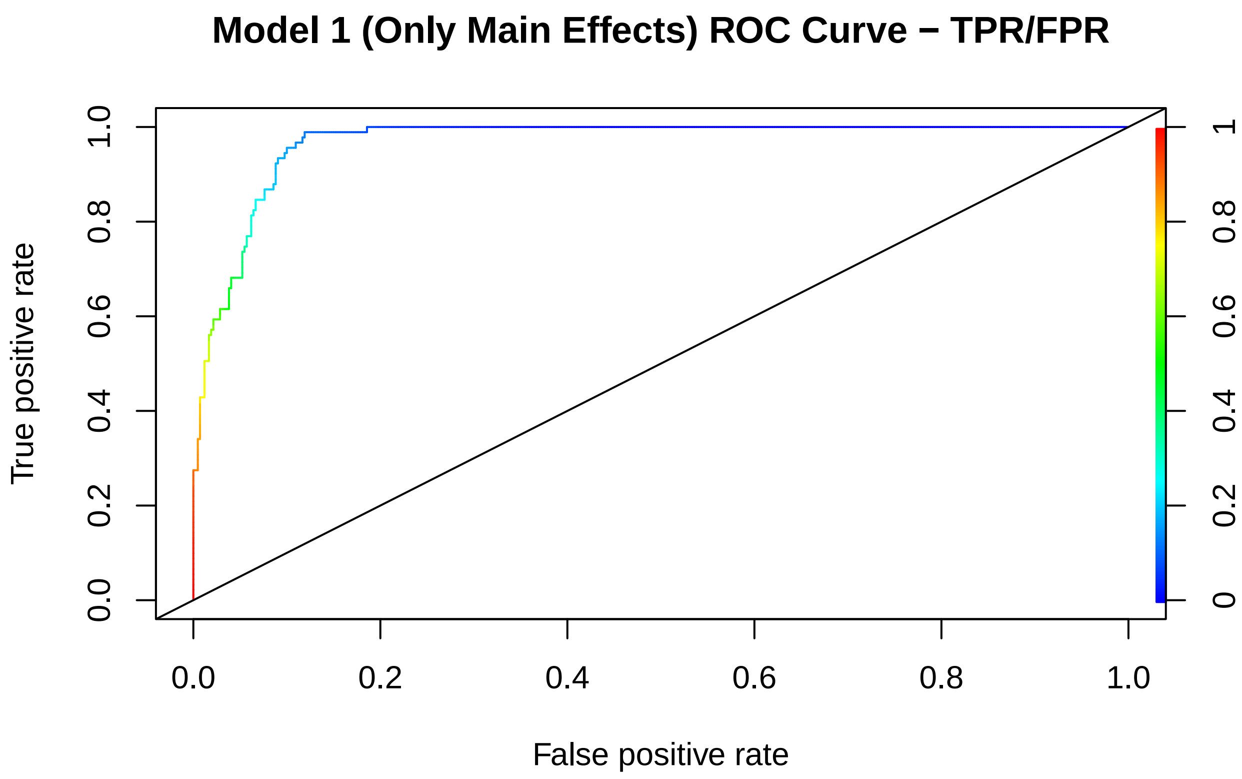
```r
# create train and test set
set.seed(213)
index = createDataPartition(depression$Depression, p = 0.80, list = FALSE, times = 1)
depression_train = depression[index,]
depression_test = depression[-index,]
```

```r
# create model with all predictors (no interaction effects)
depression_glm = glm(Depression ~ ., data = depression_train, family = "binomial")
summary(depression_glm)
```

```
##
## Call:
## glm(formula = Depression ~ ., family = "binomial", data = depression_train)
##
## Coefficients:
##                                                Estimate Std. Error z value
## (Intercept)                                     5.11463    0.48202  10.611
## GenderMale                                     -0.13997    0.19537  -0.716
## Age                                            -0.22579    0.01685 -13.403
## Working.Professional.or.StudentWorking Professional -1.71232    0.22493  -7.613
## Sleep.Duration.L                               -0.99890    0.20060  -4.980
## Sleep.Duration.Q                                0.01483    0.19541   0.076
## Sleep.Duration.C                                0.03773    0.19650   0.192
## Dietary.HabitsModerate                         -0.65864    0.23724  -2.776
## Dietary.HabitsHealthy                          -1.38810    0.24582  -5.647
## DegreeBachelors Degree                         -0.35659    0.28808  -1.238
## DegreePost-Graduate Degree                     -0.39307    0.30470  -1.290
## Work.Study.Hours                                     0.24047    0.02836   8.479
## Financial.Stress.L                                   2.19078    0.24623   8.897
## Financial.Stress.Q                                   0.04355    0.22225   0.196
## Financial.Stress.C                                  -0.11520    0.21776  -0.529
## Financial.Stress^4                                   0.09278    0.21523   0.431
## Family.History.of.Mental.IllnessYes                  0.71387    0.19667   3.630
## Pressure.L                                           3.68668    0.29334  12.568
## Pressure.Q                                          -0.27089    0.22967  -1.179
## Pressure.C                                           0.06774    0.23627   0.287
## Pressure^4                                           0.07038    0.21915   0.321
## Satisfaction.L                                      -2.86025    0.26740 -10.697
## Satisfaction.Q                                       0.02456    0.22286   0.110
## Satisfaction.C                                       0.15893    0.21990   0.723
## Satisfaction^4                                       0.26776    0.21700   1.234
##                                                     Pr(>|z|)    
## (Intercept)                                          < 2e-16 ***
## GenderMale                                          0.473705    
## Age                                                  < 2e-16 ***
## Working.Professional.or.StudentWorking Professional 2.68e-14 ***
## Sleep.Duration.L                                    6.37e-07 ***
## Sleep.Duration.Q                                    0.939487    
## Sleep.Duration.C                                    0.847750    
## Dietary.HabitsModerate                              0.005498 ** 
## Dietary.HabitsHealthy                               1.64e-08 ***
## DegreeBachelors Degree                              0.215785    
## DegreePost-Graduate Degree                          0.197042    
## Work.Study.Hours                                     < 2e-16 ***
## Financial.Stress.L                                   < 2e-16 ***
## Financial.Stress.Q                                  0.844647    
## Financial.Stress.C                                  0.596791    
## Financial.Stress^4                                  0.666436    
## Family.History.of.Mental.IllnessYes                 0.000284 ***
## Pressure.L                                           < 2e-16 ***
## Pressure.Q                                          0.238220    
## Pressure.C                                          0.774332    
## Pressure^4                                          0.748097    
## Satisfaction.L                                       < 2e-16 ***
## Satisfaction.Q                                      0.912239    
## Satisfaction.C                                      0.469859    
## Satisfaction^4                                      0.217241    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1915.51  on 2044  degrees of freedom
## Residual deviance:  717.15  on 2020  degrees of freedom
## AIC: 767.15
##
## Number of Fisher Scoring iterations: 8
```

```
# draw a roc curve for true positive rate and true negative rate to find the optimal cutoff
glm_predictions = predict(depression_glm, newdata = depression_test, type = "response")
prob_predictions = prediction(glm_predictions, depression_test$Depression)
```

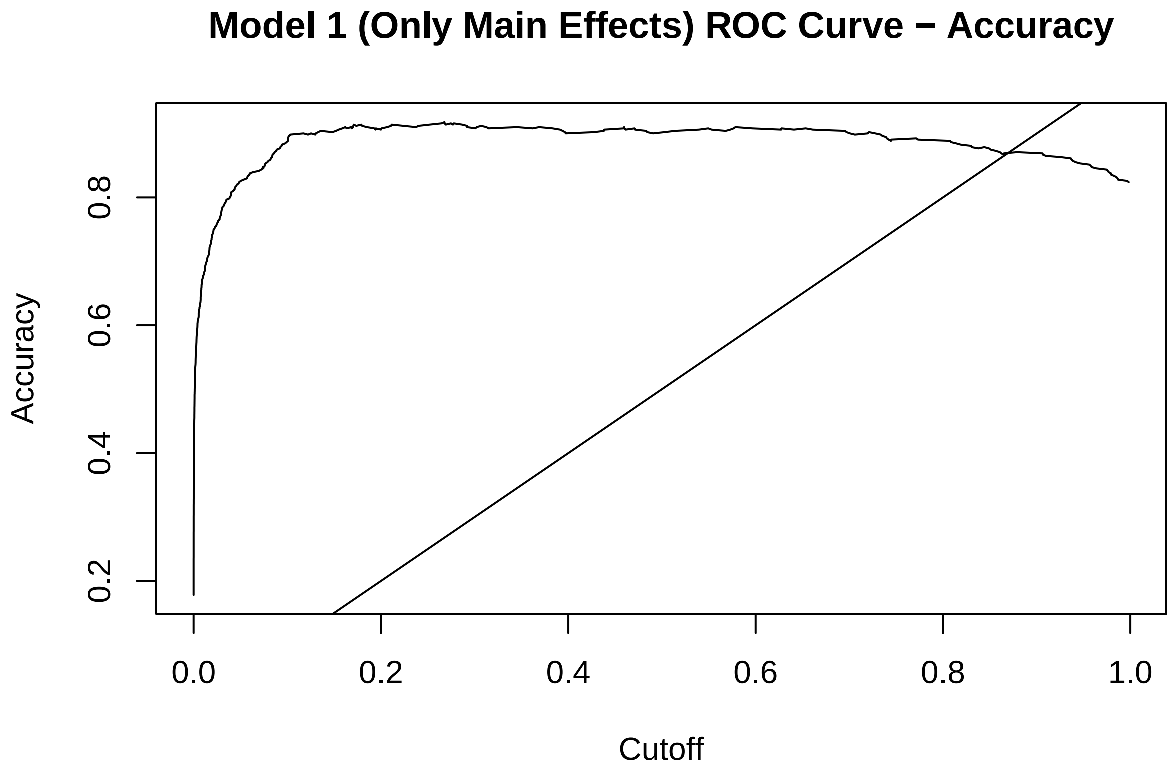
```r
roc_curve = performance(prob_predictions, "tpr", "fpr")
plot(roc_curve, colorize = TRUE, main = "Model 1 (Only Main Effects) ROC Curve - TPR/FPR")
abline(0, 1)
```



```r
# auc value
unlist(slot(performance(prob_predictions, "auc"), "y.values"))
```

```
## [1] 0.9682365
```

```r
acc = performance(prob_predictions, "acc")
plot(acc, main = "Model 1 (Only Main Effects) ROC Curve - Accuracy")
abline(0, 1)
```



```
glm_predictions2 = predict(depression_glm, newdata = depression_test)
glm_predictions2 = ifelse(glm_predictions2 > 0.30, "Yes", "No")
glm_predictions2 = as.factor(glm_predictions2)
confusionMatrix(glm_predictions2, depression_test$Depression)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  410  37
##        Yes  10  54
##
##                Accuracy : 0.908
##                  95% CI : (0.8796, 0.9316)
##     No Information Rate : 0.8219
##     P-Value [Acc > NIR] : 2.887e-08
##
##                   Kappa : 0.6445
##
##  Mcnemar's Test P-Value : 0.0001491
##
##             Sensitivity : 0.9762
##             Specificity : 0.5934
##          Pos Pred Value : 0.9172
##          Neg Pred Value : 0.8437
##              Prevalence : 0.8219
##          Detection Rate : 0.8023
##    Detection Prevalence : 0.8748
##       Balanced Accuracy : 0.7848
##
##        'Positive' Class : No
##
```

```
# create models for interaction effects of each categorical variable and see if there are any significa
# summary(glm(Depression ~ Gender*., data = depression_train, family = "binomial"))
```

```
# summary(glm(Depression ~ Working.Professional.or.Student*., data = depression_train, family = "binomi
```

```
# summary(glm(Depression ~ Sleep.Duration*., data = depression_train, family = "binomial"))
```

```
# summary(glm(Depression ~ Dietary.Habits*., data = depression_train, family = "binomial"))
```

```
# summary(glm(Depression ~ Degree*., data = depression_train, family = "binomial"))
```

```
# summary(glm(Depression ~ Work.Study.Hours*., data = depression_train, family = "binomial"))
```

```
# summary(glm(Depression ~ Financial.Stress*., data = depression_train, family = "binomial"))
```

```
# summary(glm(Depression ~ Family.History.of.Mental.Illness*., data = depression_train, family = "binom
```

```
# summary(glm(Depression ~ Pressure*., data = depression_train, family = "binomial"))
```

```
# summary(glm(Depression ~ Satisfaction*., data = depression_train, family = "binomial"))
```

None of the interaction effects were meaningfully significant; we will not be adding interaction effects to our model.

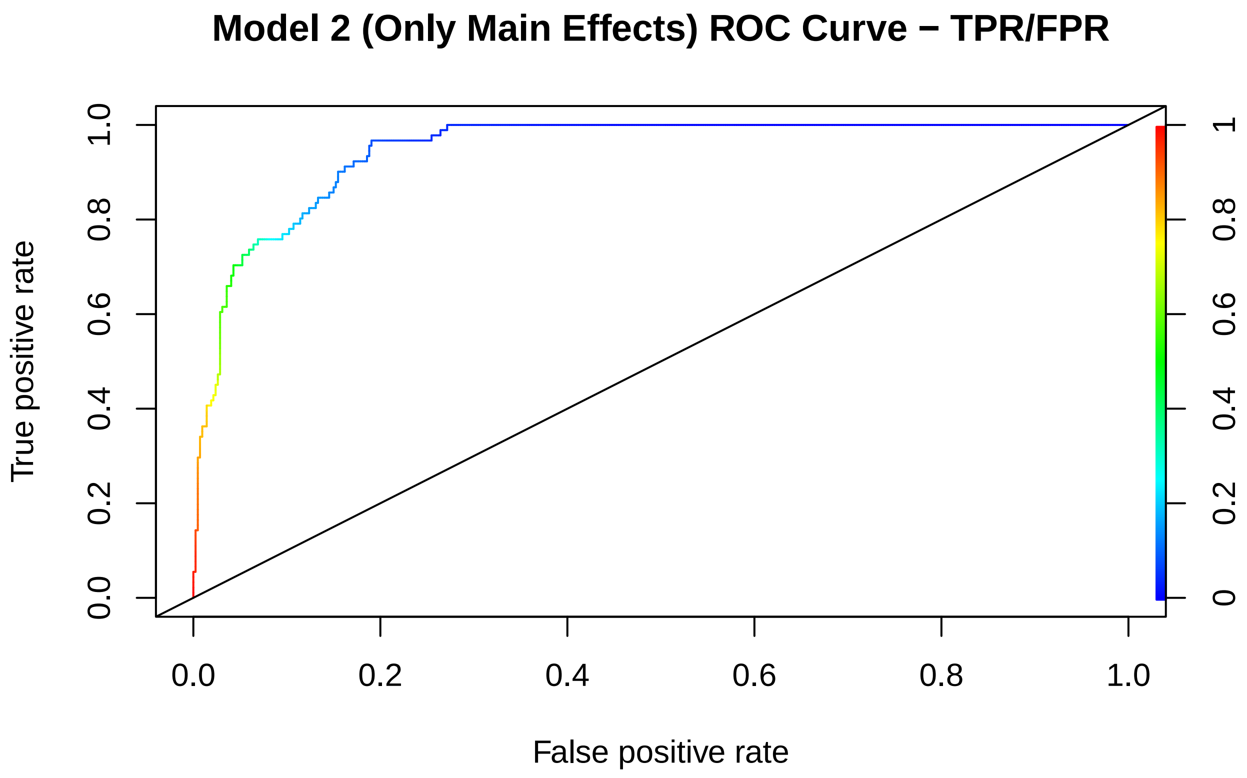
```
# create a table to easily see top important predictors and their odds for the first model
vI = cbind(varImp(depression_glm), Odds = exp(summary(depression_glm)$coefficients[-1, 1]), PValue = su
vI = vI[order(-vI$Overall), , drop = FALSE]
vI
```

```
##                                                        Overall        Odds
## Age                                                13.40322517  0.79788617
## Pressure.L                                         12.56814265 39.91217363
## Satisfaction.L                                     10.69661163  0.05725443
## Financial.Stress.L                                  8.89740988  8.94220158
## Work.Study.Hours                                    8.47936809  1.27185262
## Working.Professional.or.StudentWorking Professional  7.61277230  0.18044644
## Dietary.HabitsHealthy                               5.64675156  0.24954967
## Sleep.Duration.L                                    4.97960918  0.36828571
## Family.History.of.Mental.IllnessYes                 3.62979691  2.04188174
## Dietary.HabitsModerate                              2.77630903  0.51755473
## DegreePost-Graduate Degree                          1.29002623  0.67498267
## DegreeBachelors Degree                              1.23781538  0.70006270
## Satisfaction^4                                      1.23389757  1.30702719
## Pressure.Q                                          1.17944749  0.76270197
## Satisfaction.C                                      0.72270837  1.17225168
## GenderMale                                          0.71646445  0.86938069
## Financial.Stress.C                                  0.52902035  0.89119017
## Financial.Stress^4                                  0.43104407  1.09721520
## Pressure^4                                          0.32114983  1.07291549
## Pressure.C                                          0.28671267  1.07008876
## Financial.Stress.Q                                  0.19595350  1.04451269
## Sleep.Duration.C                                     0.19198956  1.03844568
## Satisfaction.Q                                       0.11021421  1.02486613
## Sleep.Duration.Q                                     0.07591467  1.01494523
##                                                           PValue
## Age                                                 5.789167e-41
## Pressure.L                                          3.160620e-36
## Satisfaction.L                                      1.055682e-26
## Financial.Stress.L                                  5.716507e-19
## Work.Study.Hours                                    2.264203e-17
## Working.Professional.or.StudentWorking Professional 2.682782e-14
## Dietary.HabitsHealthy                               1.635078e-08
## Sleep.Duration.L                                    6.371280e-07
## Family.History.of.Mental.IllnessYes                 2.836443e-04
## Dietary.HabitsModerate                              5.497992e-03
## DegreePost-Graduate Degree                          1.970416e-01
## DegreeBachelors Degree                              2.157845e-01
## Satisfaction^4                                      2.172411e-01
## Pressure.Q                                          2.382200e-01
## Satisfaction.C                                      4.698591e-01
## GenderMale                                          4.737046e-01
## Financial.Stress.C                                  5.967913e-01
## Financial.Stress^4                                  6.664363e-01
## Pressure^4                                          7.480968e-01
## Pressure.C                                          7.743323e-01
## Financial.Stress.Q                                  8.446466e-01
## Sleep.Duration.C                                    8.477504e-01
## Satisfaction.Q                                      9.122395e-01
## Sleep.Duration.Q                                    9.394870e-01
```

```
depression_glm2 = glm(Depression ~ Age + Pressure + Satisfaction + Work.Study.Hours + Financial.Stress,
summary(depression_glm2)
```

```
##
## Call:
## glm(formula = Depression ~ Age + Pressure + Satisfaction + Work.Study.Hours +
##     Financial.Stress, family = "binomial", data = depression_train)
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)         4.046936   0.383394  10.556   <2e-16 ***
## Age                -0.236131   0.013918 -16.966   <2e-16 ***
## Pressure.L          3.180399   0.253172  12.562   <2e-16 ***
## Pressure.Q         -0.239932   0.210775  -1.138    0.255
## Pressure.C         -0.002257   0.214985  -0.011    0.992
## Pressure^4          0.039820   0.199568   0.200    0.842
## Satisfaction.L     -2.341404   0.226697 -10.328   <2e-16 ***
## Satisfaction.Q      0.039946   0.203179   0.197    0.844
## Satisfaction.C      0.027563   0.197979   0.139    0.889
## Satisfaction^4      0.148460   0.198117   0.749    0.454
## Work.Study.Hours    0.218854   0.025688   8.520   <2e-16 ***
## Financial.Stress.L  1.860387   0.213431   8.717   <2e-16 ***
## Financial.Stress.Q  0.149191   0.200228   0.745    0.456
## Financial.Stress.C -0.139363   0.199778  -0.698    0.485
## Financial.Stress^4  0.207810   0.197858   1.050    0.294
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1915.51  on 2044  degrees of freedom
## Residual deviance:  846.31  on 2030  degrees of freedom
## AIC: 876.31
##
## Number of Fisher Scoring iterations: 7
```

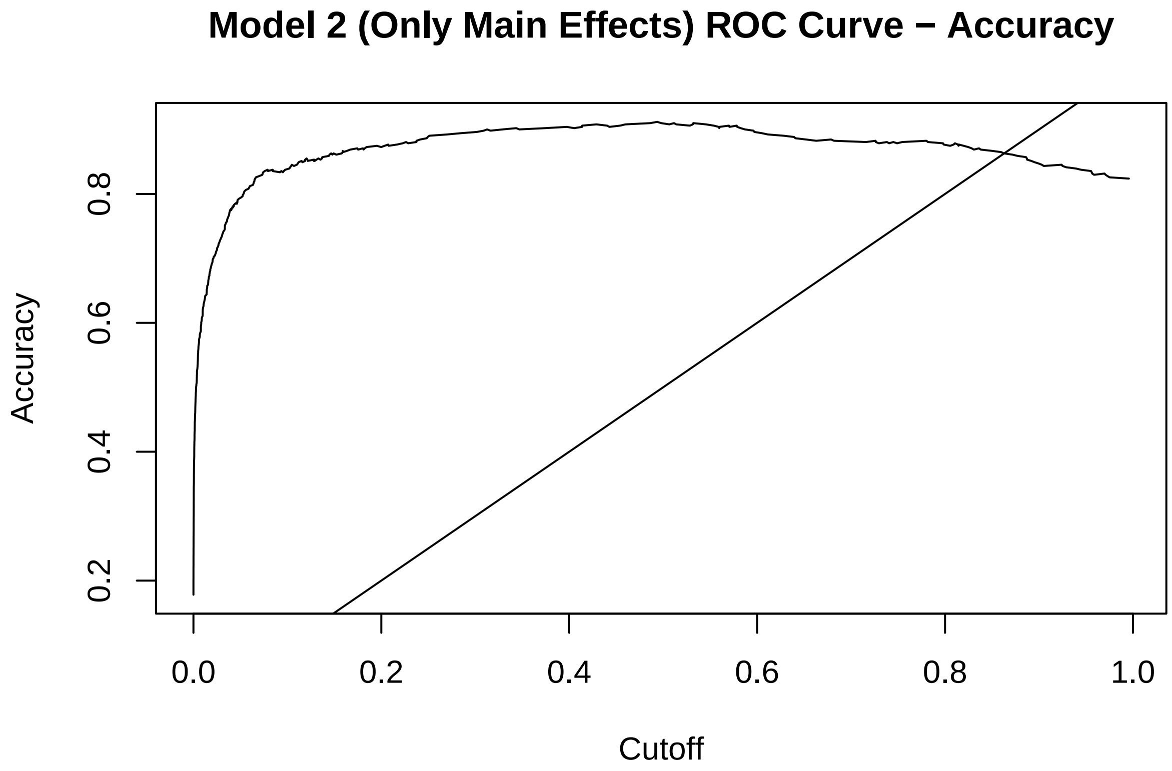
```
# draw a roc curve for true positive rate and true negative rate to find the optimal cutoff
glm_predictions3 = predict(depression_glm2, newdata = depression_test, type = "response")
prob_predictions2 = prediction(glm_predictions3, depression_test$Depression)
roc_curve2 = performance(prob_predictions2, "tpr", "fpr")
plot(roc_curve2, colorize = TRUE, main = "Model 2 (Only Main Effects) ROC Curve - TPR/FPR")
abline(0, 1)
```



```
# auc value
unlist(slot(performance(prob_predictions2, "auc"), "y.values"))
```

```
## [1] 0.946494
```

```
acc2 = performance(prob_predictions2, "acc")
plot(acc2, main = "Model 2 (Only Main Effects) ROC Curve - Accuracy")
abline(0, 1)
```



```
glm_predictions4 = predict(depression_glm2, newdata = depression_test)
glm_predictions4 = ifelse(glm_predictions4 > 0.35, "Yes", "No")
glm_predictions4 = as.factor(glm_predictions4)
confusionMatrix(glm_predictions4, depression_test$Depression)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  408  39
##        Yes  12  52
##
##                Accuracy : 0.9002
##                  95% CI : (0.8709, 0.9248)
##     No Information Rate : 0.8219
##     P-Value [Acc > NIR] : 5.264e-07
##
##                   Kappa : 0.6142
##
##  Mcnemar's Test P-Value : 0.0002719
##
##             Sensitivity : 0.9714
##             Specificity : 0.5714
##          Pos Pred Value : 0.9128
##          Neg Pred Value : 0.8125
##              Prevalence : 0.8219
##          Detection Rate : 0.7984
##    Detection Prevalence : 0.8748
##       Balanced Accuracy : 0.7714
##
##        'Positive' Class : No
##
```

```
# create a table to easily see top important predictors and their odds for the second model
vI2 = cbind(varImp(depression_glm2), Odds = exp(summary(depression_glm2)$coefficients[-1, 1]), PValue =
vI2 = vI2[order(-vI2$Overall), , drop = FALSE]
vI2
```

```
##                         Overall       Odds       PValue
## Age                 16.96601457  0.7896770 1.465465e-64
## Pressure.L          12.56222414 24.0563510 3.406212e-36
## Satisfaction.L      10.32835648  0.0961925 5.245156e-25
## Financial.Stress.L   8.71657191  6.4262231 2.867519e-18
## Work.Study.Hours     8.51971256  1.2446493 1.599498e-17
## Pressure.Q           1.13833471  0.7866813 2.549807e-01
## Financial.Stress^4   1.05029637  1.2309787 2.935819e-01
## Satisfaction^4       0.74935466  1.1600466 4.536435e-01
## Financial.Stress.Q   0.74510560  1.1608943 4.562079e-01
## Financial.Stress.C   0.69759215  0.8699119 4.854323e-01
## Pressure^4           0.19952911  1.0406231 8.418489e-01
## Satisfaction.Q       0.19660646  1.0407548 8.441355e-01
## Satisfaction.C       0.13922354  1.0279467 8.892735e-01
## Pressure.C           0.01050058  0.9977451 9.916219e-01
```

```
paste("First Model Residual Deviance: ", depression_glm$deviance)
```

```
## [1] "First Model Residual Deviance:  717.146403740301"
```

```
paste("Second Model Residual Deviance: ", depression_glm2$deviance)
```

```
## [1] "Second Model Residual Deviance:  846.31369830212"
```

```
train_control = trainControl(method = "repeatedcv", number = 10, repeats = 3, classProbs = TRUE)
depression_cvglm = train(Depression ~ .,
                         data = depression_train,
                         method = "glm",
                         family = binomial,
                         trControl = train_control)
```

```
depression_cvglm$results
```

```
##   parameter  Accuracy     Kappa AccuracySD    KappaSD
## 1      none 0.8968226 0.6369108 0.02223823 0.07530717
```

```
cvglm_predictions = predict(depression_cvglm, depression_test)
confusionMatrix(cvglm_predictions, depression_test$Depression)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  405  35
##        Yes  15  56
##
```

```
##                Accuracy : 0.9022
##                  95% CI : (0.873, 0.9265)
##     No Information Rate : 0.8219
##     P-Value [Acc > NIR] : 2.632e-07
##
##                   Kappa : 0.6343
##
##  Mcnemar's Test P-Value : 0.00721
##
##             Sensitivity : 0.9643
##             Specificity : 0.6154
##          Pos Pred Value : 0.9205
##          Neg Pred Value : 0.7887
##              Prevalence : 0.8219
##          Detection Rate : 0.7926
##    Detection Prevalence : 0.8611
##       Balanced Accuracy : 0.7898
##
##        'Positive' Class : No
##
```

```
varImp(depression_cvglm)
```

```
## glm variable importance
##
##   only 20 most important variables shown (out of 24)
##
##                                                        Overall
## Age                                                    100.000
## Pressure.L                                              93.734
## Satisfaction.L                                          79.691
## Financial.Stress.L                                      66.191
## Work.Study.Hours                                        63.054
## `Working.Professional.or.StudentWorking Professional`   56.552
## Dietary.HabitsHealthy                                   41.800
## Sleep.Duration.L                                        36.794
## Family.History.of.Mental.IllnessYes                     26.666
## Dietary.HabitsModerate                                  20.262
## `DegreePost-Graduate Degree`                             9.110
## `DegreeBachelors Degree`                                 8.718
## `Satisfaction^4`                                         8.689
## Pressure.Q                                               8.280
## Satisfaction.C                                           4.853
## GenderMale                                               4.806
## Financial.Stress.C                                       3.400
## `Financial.Stress^4`                                     2.665
## `Pressure^4`                                             1.840
## Pressure.C                                               1.582
```

```
train_control2 = trainControl(method = "repeatedcv", number = 10, repeats = 3, classProbs = TRUE)
depression_cvglm2 = train(Depression ~ Age + Pressure + Satisfaction + Work.Study.Hours + Financial.Str
                         data = depression_train,
                         method = "glm",
                         family = binomial,
                         trControl = train_control2)
```

```r
depression_cvglm2$results
```

```
##   parameter Accuracy     Kappa AccuracySD  KappaSD
## 1      none 0.895516 0.6248041 0.02400873 0.086268
```

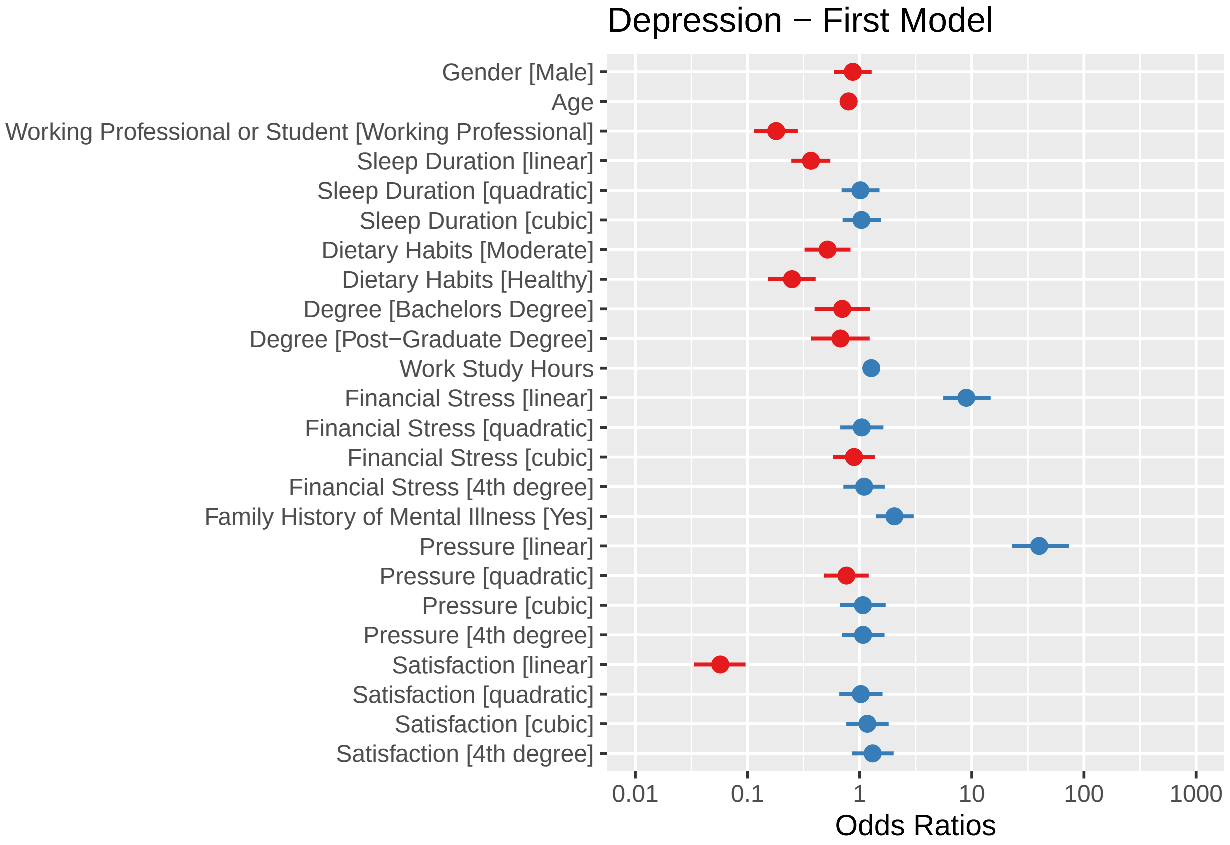
```r
cvglm_predictions2 = predict(depression_cvglm2, depression_test)
confusionMatrix(cvglm_predictions2, depression_test$Depression)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  402  29
##        Yes  18  62
##
##                Accuracy : 0.908
##                  95% CI : (0.8796, 0.9316)
##     No Information Rate : 0.8219
##     P-Value [Acc > NIR] : 2.887e-08
##
##                   Kappa : 0.6702
##
##  Mcnemar's Test P-Value : 0.1447
##
##             Sensitivity : 0.9571
##             Specificity : 0.6813
##          Pos Pred Value : 0.9327
##          Neg Pred Value : 0.7750
##              Prevalence : 0.8219
##          Detection Rate : 0.7867
##    Detection Prevalence : 0.8434
##       Balanced Accuracy : 0.8192
##
##        'Positive' Class : No
##
```

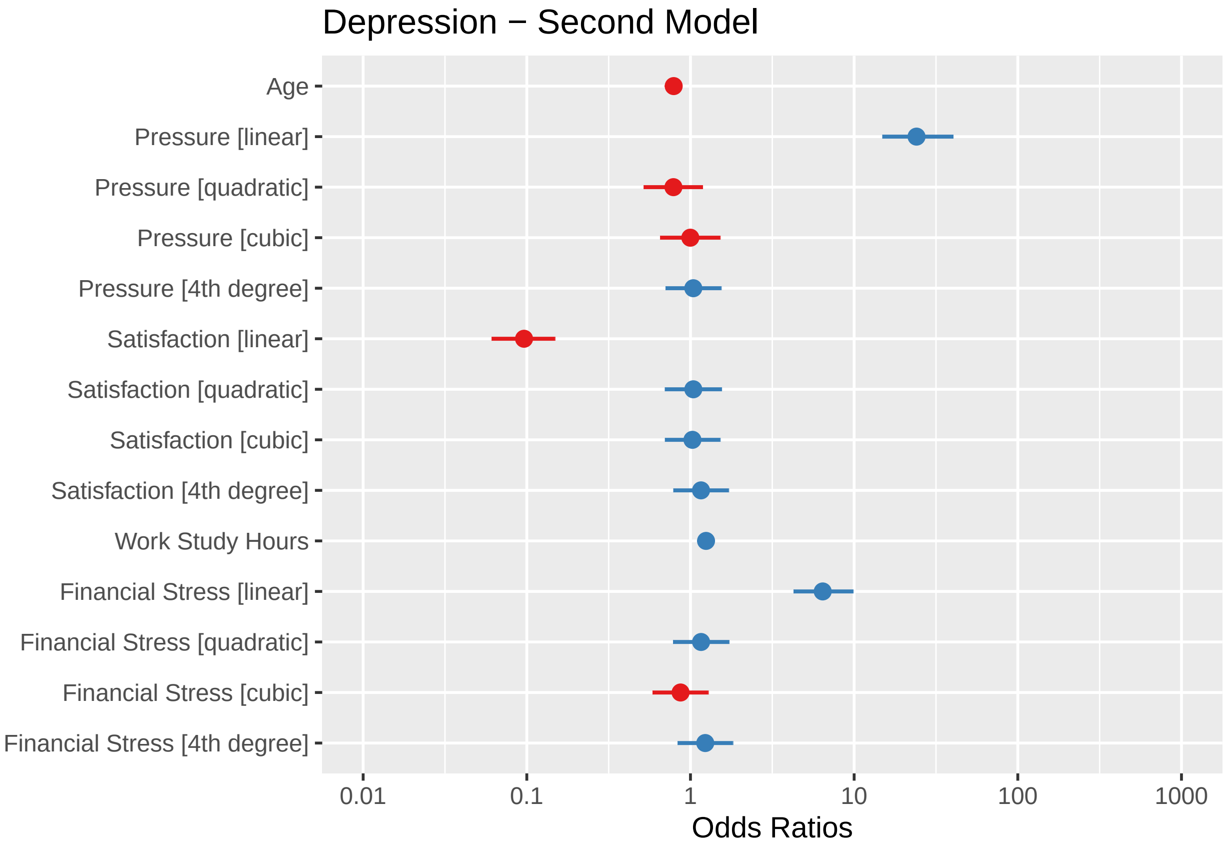
```r
varImp(depression_cvglm2)
```

```
## glm variable importance
##
##                      Overall
## Age                 100.0000
## Pressure.L           74.0274
## Satisfaction.L       60.8525
## Financial.Stress.L   51.3465
## Work.Study.Hours     50.1855
## Pressure.Q            6.6517
## `Financial.Stress^4`  6.1325
## `Satisfaction^4`      4.3576
## Financial.Stress.Q    4.3325
## Financial.Stress.C    4.0523
## `Pressure^4`          1.1148
## Satisfaction.Q        1.0976
## Satisfaction.C        0.7592
## Pressure.C            0.0000
```

```r
plot_model(depression_glm, title = "Depression - First Model")
```



```r
plot_model(depression_glm2, title = "Depression - Second Model")
```



$$\text{logit}(p) = 4.069 - 0.236 * Age + 0.219 * Work.Study.Hours + 3.180 * Pressure1 - 0.240 * Pressure2 - 0.002 * Pressure3 + 0.039 * Pre$$

$$\text{where logit}(p) = ln(\frac{p}{1-p}) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n +$$