Cardiovascular Disease Prediction with Machine Learning

Andrew Choi

2022-10-16

Contents

Introduction

This project will attempt to predict whether a patient has cardiovascular disease using different types of classification models.

What is Cardiovascular Disease?

Cardiovascular disease refers to a group of health conditions affecting the heart and blood vessels. Some examples of cardiovascular disease include coronary artery disease, pericardial disease, and cardiomyopathy. Heart and blood vessel disease is one of the leading causes of death in the United States, killing approximately 696,962 people per year. In addition, the necessary health care, medication, and premature deaths caused by coronary and blood vessel conditions cause an economic burden of $219 billion in the U.S. alone each year. Both of these statistics are directly from the CDC (Centers for Disease Control and Prevention). Cardiovascular disease is often so lethal because many people don't show symptoms of their condition until the damage is irreversible. Although the effects of cardiovascular disease are devastating to everyone, early or on-time detection by healthcare professionals have the potential to save patients who might otherwise pass away or face life-altering health conditions.

How Might This Model be Used?

As mentioned, the earlier cardiovascular disease is detected, the better the chances of survival and recovery. This model will only require arguments that are quick and easy for healthcare personnel to measure, and will try to predict with high accuracy whether or not a patient has cardiovascular disease. Although other forms of heart and blood vessel disease testing exist, they can be inaccessible for patients without health insurance or time. The aim of this model is to ensure that people who have symptoms of cardiovascular disease, patients who have close family members with a history of heart conditions, or ordinary folk simply worried about their cardiac health have the means of receiving affordable and accurate cardiovascular disease screening.

Overview of Patient Data

Data on 68,783 patients has been obtained from the following data set: https://www.kaggle.com/datasets/aiaiaidavid/cardio-data-dv13032020 There are a total of 12 variables that have been given the following descriptions:

- `AGE`: integer (years of age)
- `HEIGHT`: integer (cm)
- `WEIGHT`: integer (kg)
- `GENDER`: categorical (1: female, 2: male)
- `AP_HIGH`: systolic blood pressure, integer
- `AP_LOW`: diastolic blood pressure, integer
- `CHOLESTEROL`: categorical (1: normal, 2: above normal, 3: well above normal)
- `GLUCOSE`: categorical (1: normal, 2: above normal, 3: well above normal)
- `SMOKE`: categorical (0: no, 1: yes)
- `ALCOHOL`: categorical (0: no, 1: yes)
- `PHYSICAL_ACTIVITY`: categorical (0: no, 1: yes)
- `CARDIO_DISEASE`: categorical (0: no, 1: yes)

Loading and Cleaning Raw Data

Although the data set has already been cleaned, we can take several steps to ensure that the data and column names are as tidy as possible.

```
# Loading in the data set
cardio <- read.csv('cardiovascular_diseases_dv3 2.csv', sep = ';')
```

- Cleaning variable names

```
cardio <- cardio %>%
  clean_names()
```

- Converting height to feet, weight to pounds, and pounds to integer values
- Converting categorical variables to factors
- One-hot-encoding our factor variables
- Removing one dummy variable from each category to avoid the dummy variable trap

```
# Preparing cardio data set for use in correlation heat map
cardio_cor <- cardio

# Converting categorical variables to factors
cardio$gender <- as.factor(cardio$gender)
cardio$cholesterol <- as.factor(cardio$cholesterol)
cardio$glucose <- as.factor(cardio$glucose)
cardio$smoke <- as.factor(cardio$smoke)
cardio$alcohol <- as.factor(cardio$alcohol)
cardio$physical_activity <- as.factor(cardio$physical_activity)
cardio$cardio_disease <- as.factor(cardio$cardio_disease)

# One-hot-encoding all factors
cardio_one_hot <- one_hot(as.data.table(cardio))

cardio_one_hot <- cardio_one_hot %>%
  mutate(
    # Converting height from centimeters to feet
    height = height * 0.0328084,
    # Converting weight from kilograms to pounds
    weight = round(weight * 2.20462),
```

```
    # Converting new one-hot-encoded variables to factors
    cardio_disease_1 = factor(cardio_disease_1),
    gender_2 = factor(gender_2),
    cholesterol_2 = factor(cholesterol_2),
    cholesterol_3 = factor(cholesterol_3),
    glucose_2 = factor(glucose_2),
    glucose_3 = factor(glucose_3)
  )


# Removing one dummy variable from each category
cardio_one_hot[ , c('gender_1', 'cholesterol_1', 'glucose_1', 'smoke_0', 'alcohol_0', 'physical_activity
```

Exploratory Data Analysis

Although it's not likely that any single variable in our data set directly causes or prevents cardiovascular disease, there may be a link between certain variables and coronary disease. We will visualize these links through a correlation plot, as well as individual barplots of our predictor variables against `cardio_disease`

Correlation Plot

change ***

Although it's not likely that any single variable in our data set directly causes or prevents cardiovascular disease, there may be a link between certain variables and coronary health. We can create the equivalent of a correlation plot between our numeric and categorical variables since we've one-hot-encoded our categorical variables. This heat map can help visualize the connection between each variable.

```
cor(cardio_cor, use = 'all.obs') %>%
  ggcorrplot(colors = c('#404080', 'white',  '#69b3a2'), lab = TRUE, lab_size = 2)
```
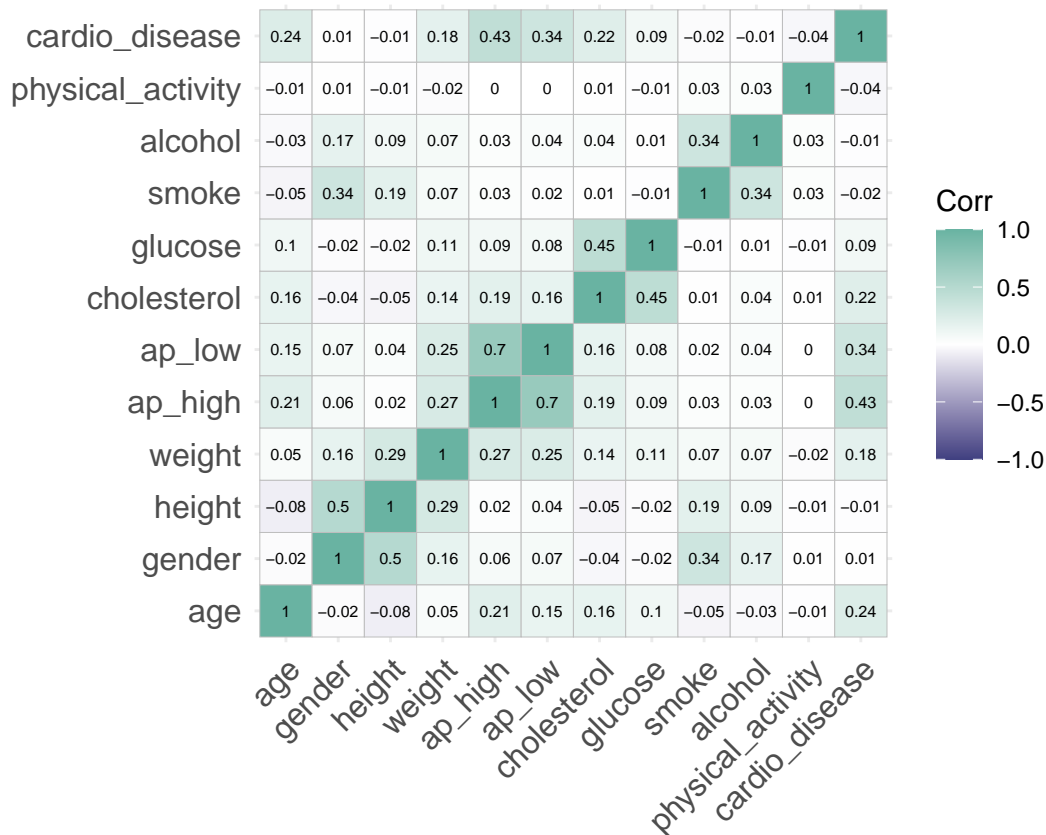
Many of the results of this heat map show connections that were expected. For example, the fact that age, weight, diastolic and systolic blood pressure, and high cholesterol levels are all positively correlated with cardiovascular disease is unsurprising. However, there are also a few results that are a bit more interesting. Smoking, drinking alcohol, and exercise all have close to no correlation with cardiovascular disease. This can be attributed to a variety of factors, one of which is the fact that `smoke`, `alcohol`, and `physical_activity` are all binary variables. This implies that someone who smokes a pack of Marlboro reds a day, drinks enough alcohol to score an integer value BAC percentage, and lives an extremely sedentary lifestyle is given the same score as someone who smokes as rarely as they drink, and chooses not to exercise. Another possible factor is that every person in this data set drinks, smokes, and exercises so moderately that their cardiovascular health is not affected. We can continue to explore the relationships between our predictors and cardiovascular disease by generating histograms and barplots between these variables.

```r
# Creating data frame that contains 'age' and 'cardio_disease' if 'cardio_disease' is 0
cardio_disease_age_0 <- select(
  cardio[cardio$cardio_disease == 0, ],
  c('age', 'cardio_disease')
  )


cardio_disease_age_0$disease <- 'Patients With Cardiovascular Disease'


# Creating data frame that contains 'age' and 'cardio_disease' if 'cardio_disease' is 1
cardio_disease_age_1 <- select(
  cardio[cardio$cardio_disease == 1, ],
```

```
  c('age', 'cardio_disease')
  )


cardio_disease_age_1$disease <- 'Patients Without Cardiovascular Disease'


# Combining the new data frames
cardio_age <- rbind(
  cardio_disease_age_0,
  cardio_disease_age_1
  )


# Plotting the data side by side
ggplot(cardio_age, aes(x = age)) +
  geom_bar(stat = 'count', aes(fill = disease), position = 'dodge', alpha = .8) +
  scale_fill_manual(values = c('#404080', '#69b3a2'), name = '') +
  labs(x = 'Age (years)', y = 'Total')
```


Age
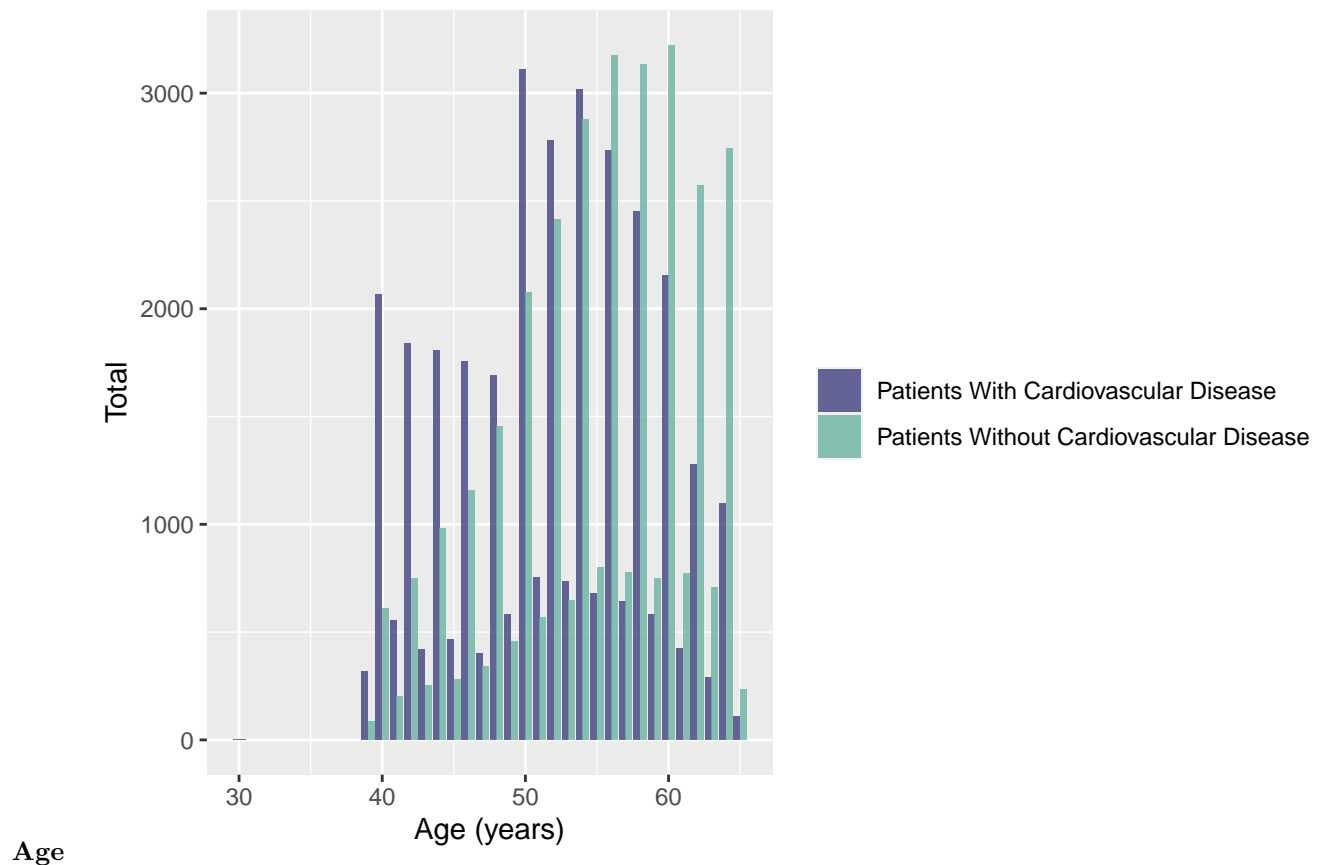
Weight   Extreme values of weight don't appear to have higher or lower rates of cardiovascular disease when compared to the average values of weight. In fact, we can see that

```
# Creating data frame that contains 'weight' and 'cardio_disease' if 'cardio_disease' is 0
cardio_disease_weight_0 <- select(
```

```r
  cardio[cardio$cardio_disease == 0,],
  c('weight' ,'cardio_disease')
)

cardio_disease_weight_0$disease <- 'Patients With Cardiovascular Disease'


# Creating data frame that contains 'weight' and 'cardio_disease' if 'cardio_disease' is 1
cardio_disease_weight_1 <- select(
  cardio[cardio$cardio_disease == 1,],
  c('weight', 'cardio_disease')
)

cardio_disease_weight_1$disease <- 'Patients Without Cardiovascular Disease'


# Combining the new data frames
cardio_weight <- rbind(
  cardio_disease_weight_0,
  cardio_disease_weight_1
)


# Plotting the data
ggplot(cardio_weight, aes(x = weight, fill = disease), alpha = .8) +
  geom_bar() +
  labs(x = 'Weight (lbs)', y = 'Total') +
  scale_fill_manual(values = c('#404080', '#69b3a2'), name = '')
```
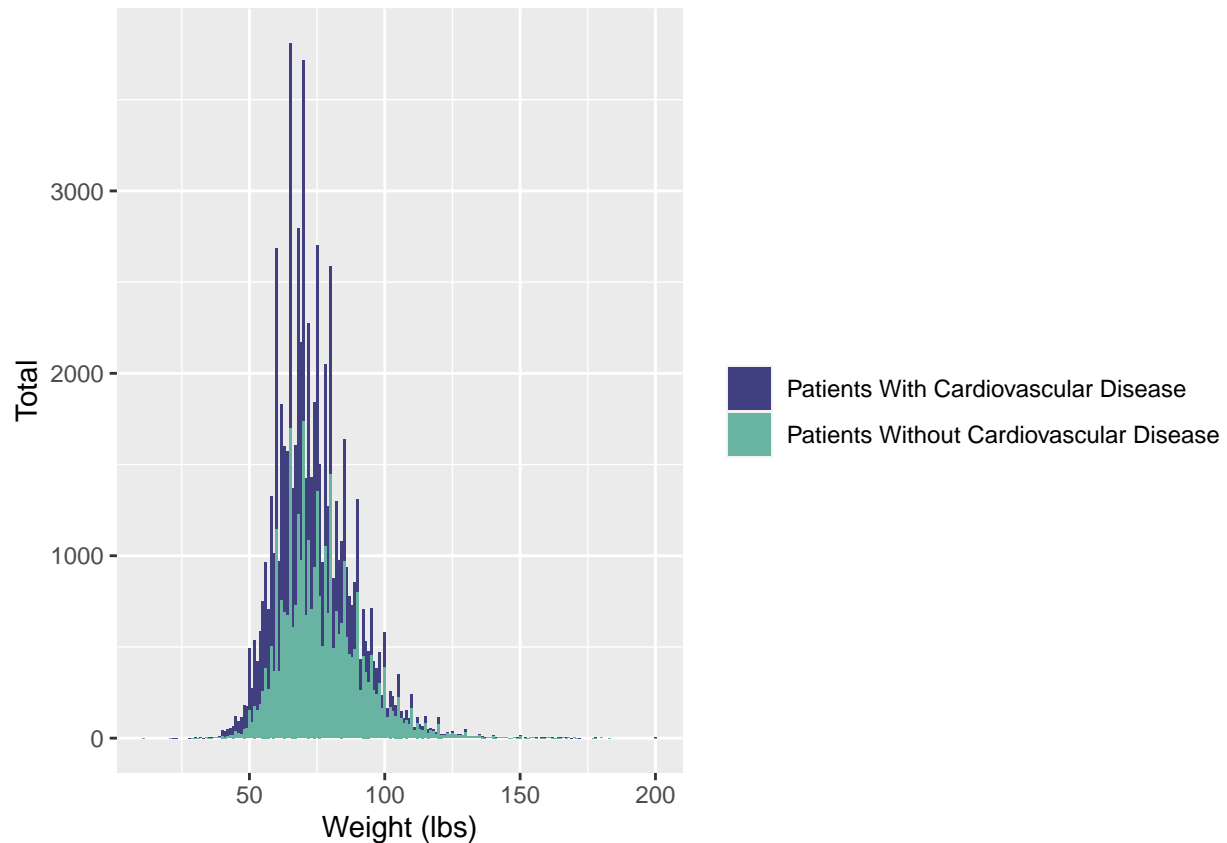
```r
# Creating data frame that contains 'ap_high' and 'cardio_disease' if 'cardio_disease' is 0
cardio_disease_ap_high_0 <- select(
  cardio[cardio$cardio_disease == 0,],
  c('ap_high' ,'cardio_disease')
)

cardio_disease_ap_high_0$disease <- 'Patients With Cardiovascular Disease'


# Creating data frame that contains 'ap_high' and 'cardio_disease' if 'cardio_disease' is 1
cardio_disease_ap_high_1 <- select(
  cardio[cardio$cardio_disease == 1,],
  c('ap_high', 'cardio_disease')
)

cardio_disease_ap_high_1$disease <- 'Patients Without Cardiovascular Disease'


# Combining the new data frames
cardio_ap_high <- rbind(
  cardio_disease_ap_high_0,
  cardio_disease_ap_high_1
)
```
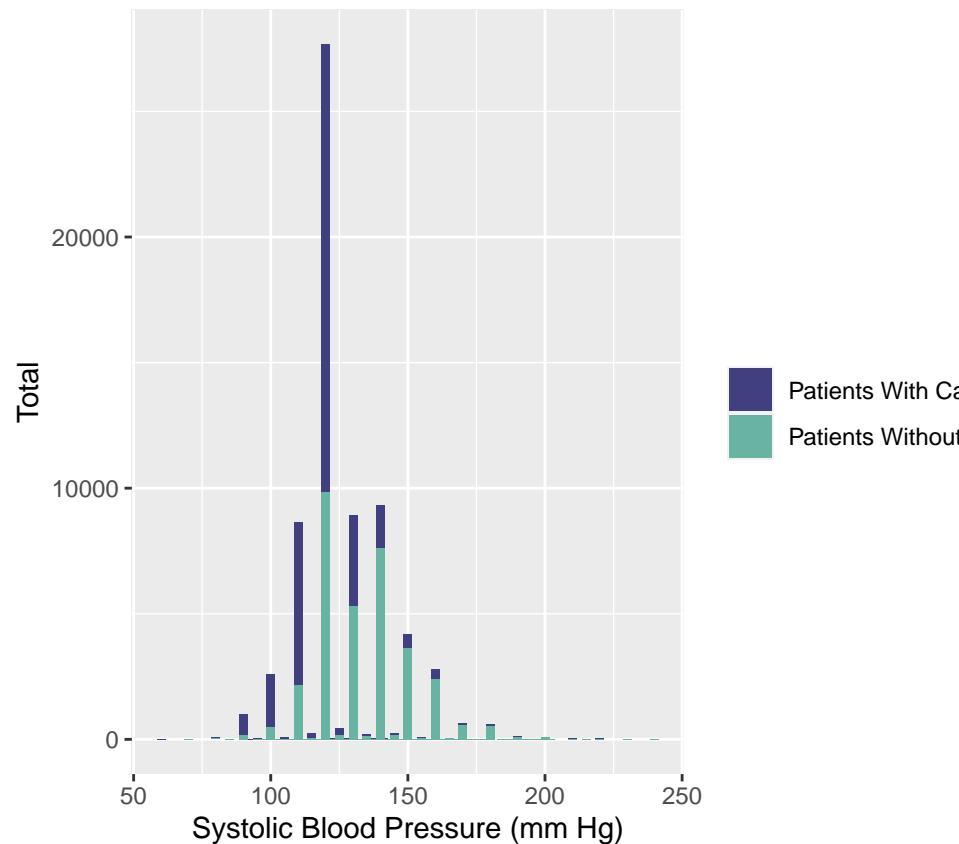
```
# Plotting the data
ggplot(cardio_ap_high, aes(x = ap_high, fill = disease), alpha = .7) +
  geom_bar(width = 3) +
  labs(x = 'Systolic Blood Pressure (mm Hg)', y = 'Total') +
  scale_fill_manual(values = c('#404080', '#69b3a2'), name = '')
```


AP High (Systolic Blood Pressure)

```
# Creating data frame that contains 'ap_low' and 'cardio_disease' if 'cardio_disease' is 0
cardio_disease_ap_low_0 <- select(
  cardio[cardio$cardio_disease == 0,],
  c('ap_low' ,'cardio_disease')
)

cardio_disease_ap_low_0$disease <- 'Patients With Cardiovascular Disease'


# Creating data frame that contains 'ap_low' and 'cardio_disease' if 'cardio_disease' is 1
cardio_disease_ap_low_1 <- select(
  cardio[cardio$cardio_disease == 1,],
  c('ap_low', 'cardio_disease')
)
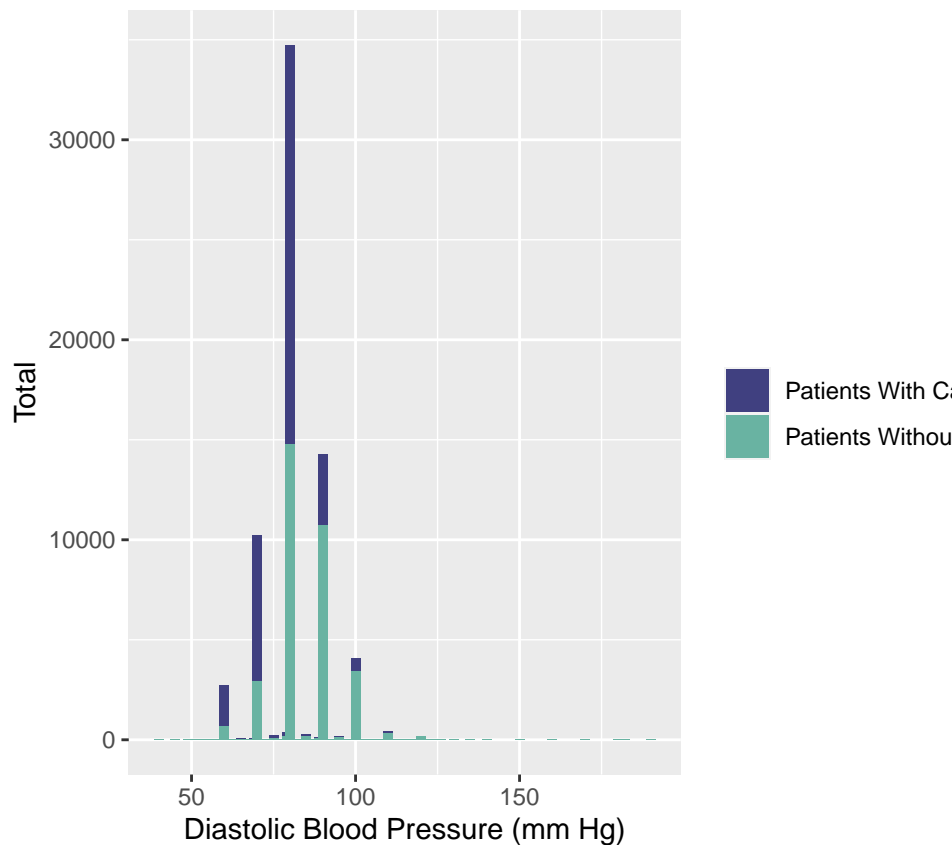```

```
cardio_disease_ap_low_1$disease <- 'Patients Without Cardiovascular Disease'


# Combining the new data frames
cardio_ap_low <- rbind(
  cardio_disease_ap_low_0,
  cardio_disease_ap_low_1
)


# Plotting the data
ggplot(cardio_ap_low, aes(x = ap_low, fill = disease), alpha = .7) +
  geom_bar(width = 3) +
  labs(x = 'Diastolic Blood Pressure (mm Hg)', y = 'Total') +
  scale_fill_manual(values = c('#404080', '#69b3a2'), name = '')
```


AP Low (Diastolic Blood Pressure)

```
# Creating data frame that contains 'cholesterol' and 'cardio_disease' if 'cardio_disease' is 0 and 'ch
cardio_disease_cholesterol_0_1 <- select(
  cardio[cardio$cardio_disease == 0 & cardio$cholesterol == 1, ],
  c('cholesterol', 'cardio_disease')
)
```

```r
cardio_disease_cholesterol_0_1$disease <- 'Patients With Normal Cholesterol Levels Without Cardiovascula


# Creating data frame that contains 'cholesterol' and 'cardio_disease' if 'cardio_disease' is 0 and 'ch
cardio_disease_cholesterol_0_2 <- select(
  cardio[cardio$cardio_disease == 0 & cardio$cholesterol == 2, ],
  c('cholesterol', 'cardio_disease')
)

cardio_disease_cholesterol_0_2$disease <- 'Patients With Above Normal Cholesterol Levels Without Cardiov


# Creating data frame that contains 'cholesterol' and 'cardio_disease' if 'cardio_disease' is 0 and 'ch
cardio_disease_cholesterol_0_3 <- select(
  cardio[cardio$cardio_disease == 0 & cardio$cholesterol == 3, ],
  c('cholesterol', 'cardio_disease')
)

cardio_disease_cholesterol_0_3$disease <- 'Patients With Well Above Normal Cholesterol Levels Without Ca


# Creating data frame that contains 'cholesterol' and 'cardio_disease' if 'cardio_disease' is 1 and 'ch
cardio_disease_cholesterol_1_1 <- select(
  cardio[cardio$cardio_disease == 1 & cardio$cholesterol == 1, ],
  c('cholesterol', 'cardio_disease')
)

cardio_disease_cholesterol_1_1$disease <- 'Patients With Normal Cholesterol Levels With Cardiovascular I


# Creating data frame that contains 'cholesterol' and 'cardio_disease' if 'cardio_disease' is 1 and 'ch
cardio_disease_cholesterol_1_2 <- select(
  cardio[cardio$cardio_disease == 1 & cardio$cholesterol == 2, ],
  c('cholesterol', 'cardio_disease')
)

cardio_disease_cholesterol_1_2$disease <- 'Patients With Above Normal Cholesterol Levels With Cardiovasc


# Creating data frame that contains 'cholesterol' and 'cardio_disease' if 'cardio_disease' is 1 and 'ch
cardio_disease_cholesterol_1_3 <- select(
  cardio[cardio$cardio_disease == 1 & cardio$cholesterol == 3, ],
  c('cholesterol', 'cardio_disease')
)

cardio_disease_cholesterol_1_3$disease <- 'Patients With Well Above Normal Cholesterol Levels With Cardi


# Combining the new data frames
cardio_cholesterol <- rbind(
  cardio_disease_cholesterol_0_1,
  cardio_disease_cholesterol_0_2,
  cardio_disease_cholesterol_0_3,
```

```
  cardio_disease_cholesterol_1_1,
  cardio_disease_cholesterol_1_2,
  cardio_disease_cholesterol_1_3)

cardio_cholesterol$disease <- factor(cardio_cholesterol$disease, levels = c('Patients With Normal Choles

# Plotting the data
ggplot(cardio_cholesterol, aes(x = cholesterol)) +
  geom_bar(stat = 'count', aes(fill = disease), position = 'dodge', alpha = .6) +
  labs(x = 'Cholesterol levels', y = 'Total') +
  scale_fill_manual(values = c('#404080', '#69B3A2', '#6699CC', '#B8BCFF', '#FBD9E6', '#F6FBE4'), name =
```


Cholesterol
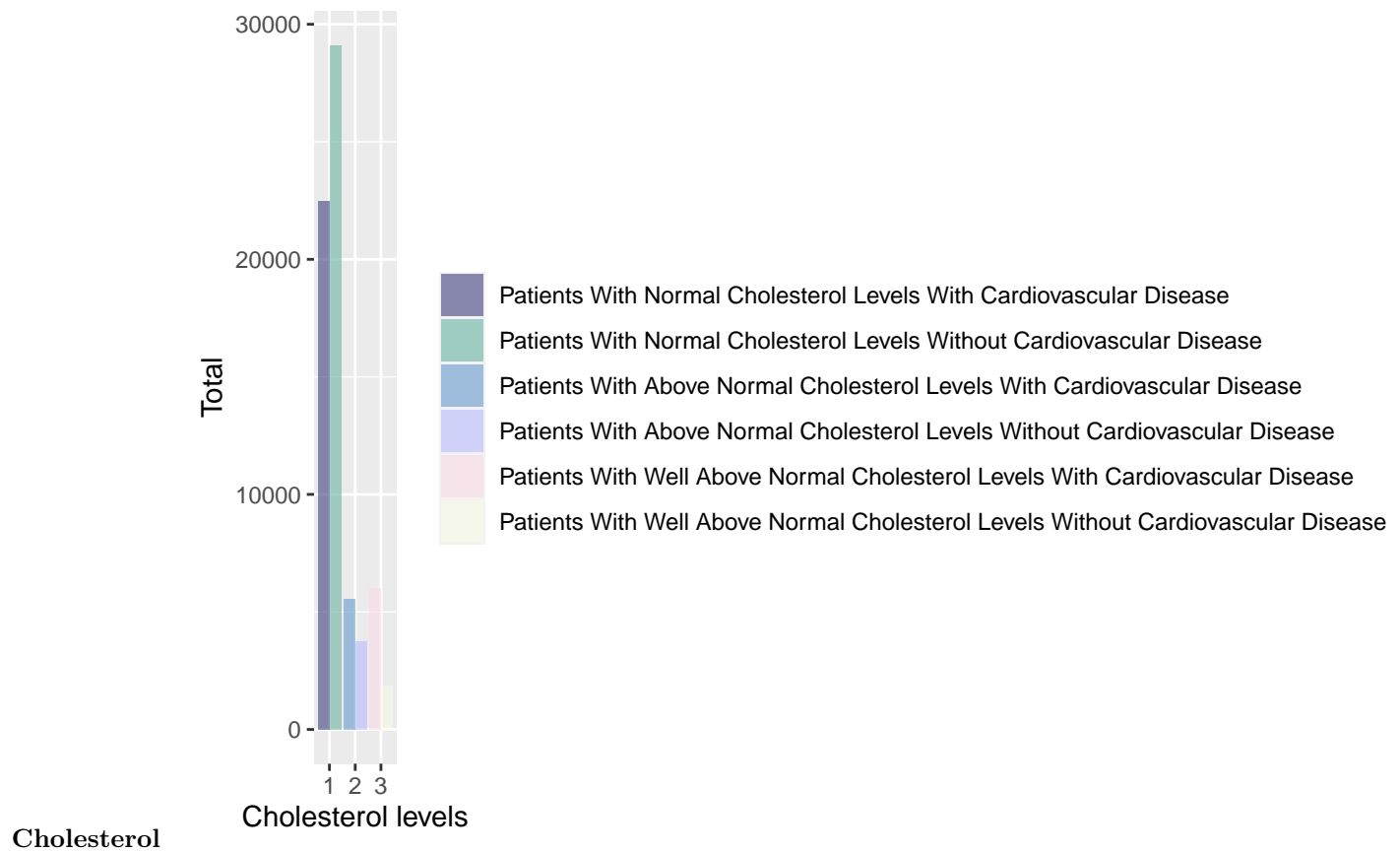
```
# Creating data frame that contains 'smoke' and 'cardio_disease' if 'smoke' is 0 and 'cardio_disease' i
cardio_disease_smoke_0_0 <- select(
  cardio[cardio$smoke == 0 & cardio$cardio_disease == 0,],
  c('smoke' ,'cardio_disease')
)

cardio_disease_smoke_0_0$disease <- 'Patients Who Do Not Smoke Without Cardiovascular Disease'
```

```r
# Creating data frame that contains 'smoke' and 'cardio_disease' if 'smoke' is 0 'cardio_disease' is 1
cardio_disease_smoke_0_1 <- select(
  cardio[cardio$smoke == 0 & cardio$cardio_disease == 1,],
  c('smoke' ,'cardio_disease')
)

cardio_disease_smoke_0_1$disease <- 'Patients Who Do Not Smoke With Cardiovascular Disease'


# Creating data frame that contains 'smoke' and 'cardio_disease' if 'smoke' is 1 'cardio_disease' is 0
cardio_disease_smoke_1_0 <- select(
  cardio[cardio$smoke == 1 & cardio$cardio_disease == 0,],
  c('smoke' ,'cardio_disease')
)

cardio_disease_smoke_1_0$disease <- 'Patients Who Smoke Without Cardiovascular Disease'


# Creating data frame that contains 'smoke' and 'cardio_disease' if 'smoke' is 1 'cardio_disease' is 1
cardio_disease_smoke_1_1 <- select(
  cardio[cardio$smoke == 1 & cardio$cardio_disease == 1,],
  c('smoke' ,'cardio_disease')
)

cardio_disease_smoke_1_1$disease <- 'Patients Who Smoke With Cardiovascular Disease'


# Combining the new data frames
cardio_smoke <- rbind(
  cardio_disease_smoke_0_0,
  cardio_disease_smoke_0_1,
  cardio_disease_smoke_1_0,
  cardio_disease_smoke_1_1
)


# Plotting the data
ggplot(cardio_smoke, aes(x = smoke)) +
  geom_bar(stat = 'count', aes(fill = disease), position = 'dodge', alpha = .7) +
  labs(x = 'Nonsmokers vs. Smokers', y = 'Total') +
  scale_fill_manual(values = c('#404080', '#69B3A2', '#6699CC', '#B8BCFF'), name = '')
```
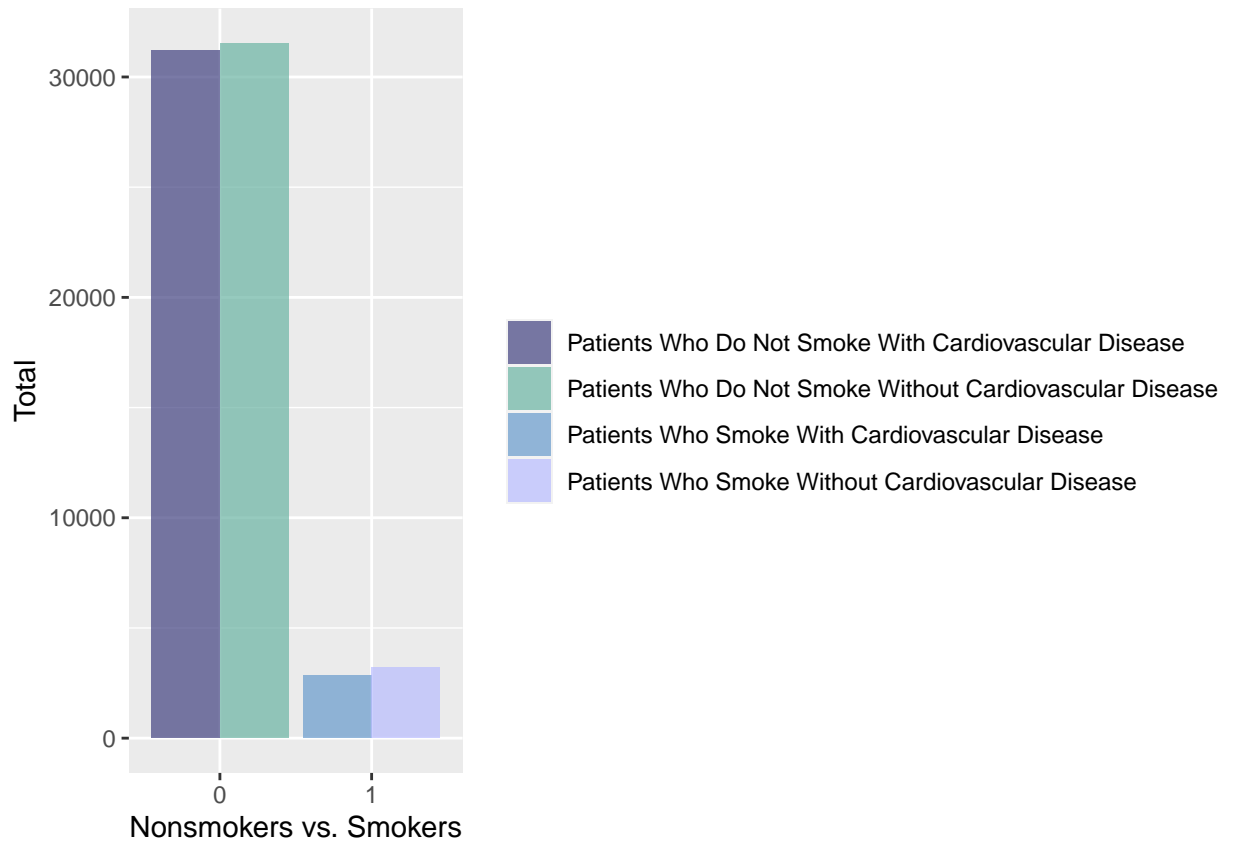
Smoke

```r
# Creating data frame that contains 'alcohol' and 'cardio_disease' if 'alcohol' is 0 and 'cardio_diseas
cardio_disease_alcohol_0_0 <- select(
  cardio[cardio$alcohol == 0 & cardio$cardio_disease == 0,],
  c('alcohol' ,'cardio_disease')
)

cardio_disease_alcohol_0_0$disease <- 'Patients Who Do Not Drink Alcohol Without Cardiovascular Disease


# Creating data frame that contains 'alcohol' and 'cardio_disease' if 'alcohol' is 0 'cardio_disease' i
cardio_disease_alcohol_0_1 <- select(
  cardio[cardio$alcohol == 0 & cardio$cardio_disease == 1,],
  c('alcohol' ,'cardio_disease')
)

cardio_disease_alcohol_0_1$disease <- 'Patients Who Do Not Drink Alcohol With Cardiovascular Disease'


# Creating data frame that contains 'alcohol' and 'cardio_disease' if 'alcohol' is 1 'cardio_disease' i
cardio_disease_alcohol_1_0 <- select(
  cardio[cardio$alcohol == 1 & cardio$cardio_disease == 0,],
  c('alcohol' ,'cardio_disease')
)
```

13

```r
cardio_disease_alcohol_1_0$disease <- 'Patients Who Drink Alcohol Without Cardiovascular Disease'


# Creating data frame that contains 'alcohol' and 'cardio_disease' if 'alcohol' is 1 'cardio_disease' i.
cardio_disease_alcohol_1_1 <- select(
  cardio[cardio$alcohol == 1 & cardio$cardio_disease == 1,],
  c('alcohol' ,'cardio_disease')
)

cardio_disease_alcohol_1_1$disease <- 'Patients Who Drink Alcohol With Cardiovascular Disease'


# Combining the new data frames
cardio_alcohol <- rbind(
  cardio_disease_alcohol_0_0,
  cardio_disease_alcohol_0_1,
  cardio_disease_alcohol_1_0,
  cardio_disease_alcohol_1_1
)


# Plotting the data
ggplot(cardio_alcohol, aes(x = alcohol)) +
  geom_bar(stat = 'count', aes(fill = disease), position = 'dodge', alpha = .7) +
  labs(x = 'Patients Who Drink vs. Patients Who Abstain', y = 'Total') +
  scale_fill_manual(values = c('#404080', '#69B3A2', '#6699CC', '#B8BCFF'), name = '')
```
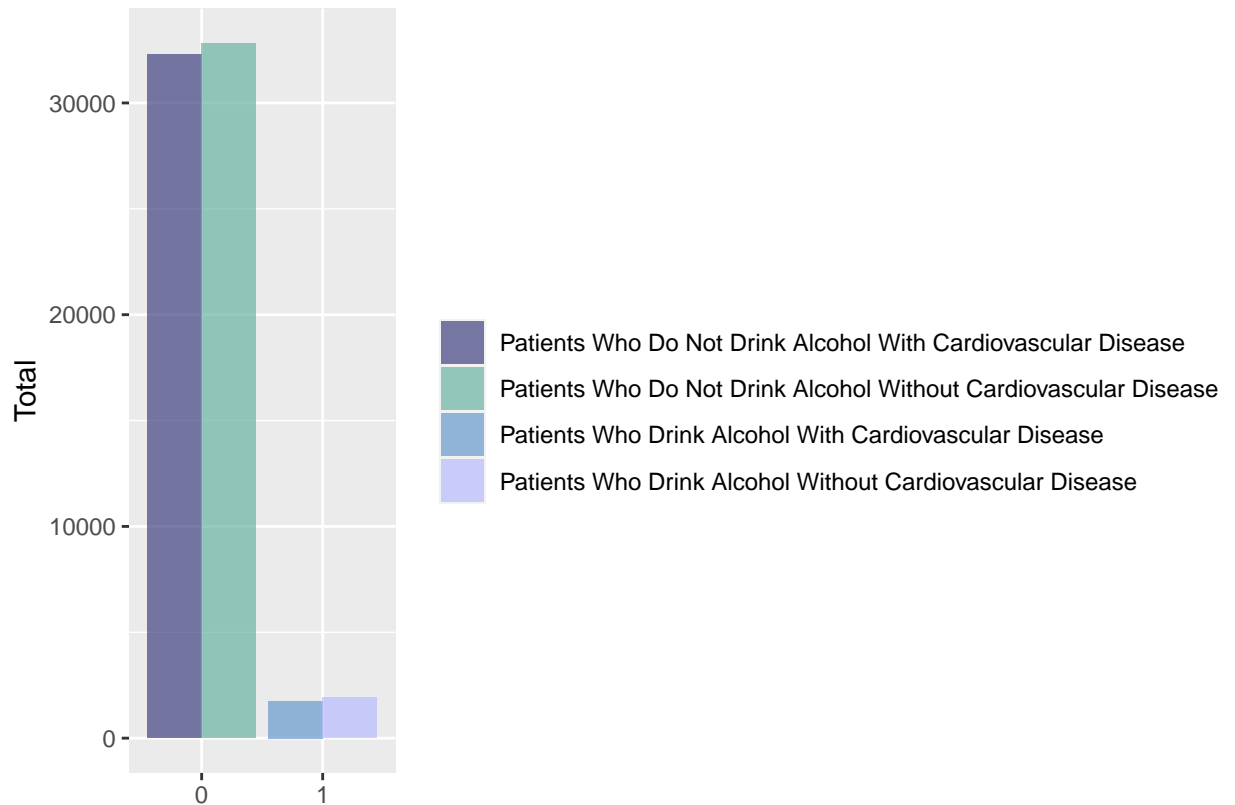
Alcohol

Patients Who Drink vs. Patients Who Abstain

```r
# Creating data frame that contains 'physical_activity' and 'cardio_disease' if 'physical_activity' is
cardio_disease_p_a_0_0 <- select(
  cardio[cardio$physical_activity == 0 & cardio$cardio_disease == 0,],
  c('physical_activity' ,'cardio_disease')
)

cardio_disease_p_a_0_0$disease <- 'Patients Who Are Not Physically Active Without Cardiovascular Disease

# Creating data frame that contains 'physical_activity' and 'cardio_disease' if 'physical_activity' is
cardio_disease_p_a_0_1 <- select(
  cardio[cardio$physical_activity == 0 & cardio$cardio_disease == 1,],
  c('physical_activity' ,'cardio_disease')
)

cardio_disease_p_a_0_1$disease <- 'Patients Who Are Not Physically Active With Cardiovascular Disease'

# Creating data frame that contains 'physical_activity' and 'cardio_disease' if 'physical_activity' is
cardio_disease_p_a_1_0 <- select(
  cardio[cardio$physical_activity == 1 & cardio$cardio_disease == 0,],
  c('physical_activity' ,'cardio_disease')
)
```

```r
cardio_disease_p_a_1_0$disease <- 'Patients Who Are Physically Active Without Cardiovascular Disease'


# Creating data frame that contains 'physical_activity' and 'cardio_disease' if 'physical_activity' is
cardio_disease_p_a_1_1 <- select(
  cardio[cardio$physical_activity == 1 & cardio$cardio_disease == 1,],
  c('physical_activity' ,'cardio_disease')
)

cardio_disease_p_a_1_1$disease <- 'Patients Who Are Physically Active With Cardiovascular Disease'


# Combining the new data frames
cardio_p_a <- rbind(
  cardio_disease_p_a_0_0,
  cardio_disease_p_a_0_1,
  cardio_disease_p_a_1_0,
  cardio_disease_p_a_1_1
)


# Plotting the data
ggplot(cardio_p_a, aes(x = physical_activity)) +
  geom_bar(stat = 'count', aes(fill = disease), position = 'dodge', alpha = .7) +
  labs(x = '', y = 'Total') +
  scale_fill_manual(values = c('#404080', '#69B3A2', '#6699CC', '#B8BCFF'), name = '')
```
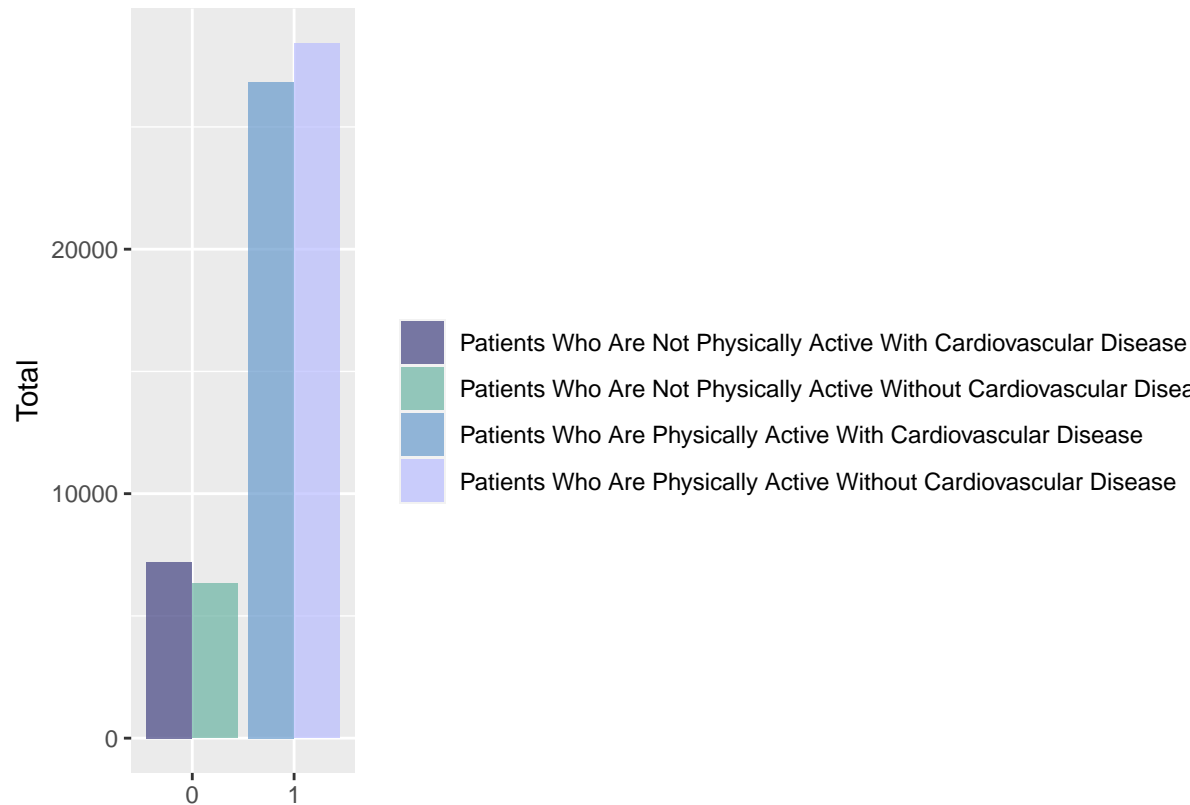
Physical Activity

Summary   Based on the fact that the histograms of `smoke`, drinking and physical activity as well as the correlation = 0, we remoce these variables from our analysis

Creating the Model

The process of creating a model begins by first splitting the data into training and testing sets. Because the data set we're working with has a large number of observations, we can afford to split it into 90% training and 10% testing without worrying that there won't be enough observations in the testing set. We will also stratify the split on our response variable, `cardio_disease`

```
cardio_split <- cardio_one_hot %>%
  initial_split(strata = cardio_disease_1, prop = .90)

cardio_train <- training(cardio_split) %>% as_tibble()
cardio_train[ ,c('smoke_1', 'alcohol_1', 'physical_activity_1')] <- list(NULL)

cardio_test <- testing(cardio_split)
```

Creating the Recipe

```
#lda, qda, naive bayes, logistic reg, boosted, svm
# good
```

```
cardio_recipe <- recipe(cardio_disease_1 ~ age + gender_2 + height + weight + ap_high + ap_low + choles
  step_dummy(gender_2) %>%
  step_dummy(cholesterol_2) %>%
  step_dummy(cholesterol_3) %>%
  step_dummy(glucose_2) %>%
  step_dummy(glucose_3) %>%
  step_normalize(all_predictors())


cardio_folds <- vfold_cv(cardio_train, v = 10, strata = cardio_disease_1)


cardio_lda <- discrim_linear() %>%
  set_mode('classification') %>%
  set_engine('MASS')

cardio_lda_workflow <- workflow() %>%
  add_model(cardio_lda) %>%
  add_recipe(cardio_recipe)

cardio_lda_fit <- fit(cardio_lda_workflow, cardio_train)


predict(cardio_lda_fit, new_data = cardio_train, type = 'prob')
```

```
## # A tibble: 61,903 x 2
##     .pred_0 .pred_1
##       <dbl>   <dbl>
##  1    0.798  0.202
##  2    0.905  0.0948
##  3    0.435  0.565
##  4    0.159  0.841
##  5    0.810  0.190
##  6    0.794  0.206
##  7    0.496  0.504
##  8    0.695  0.305
##  9    0.801  0.199
## 10    0.617  0.383
## # ... with 61,893 more rows
```

```
augment(cardio_lda_fit, new_data = cardio_train) %>%
  conf_mat(truth = cardio_disease_1, estimate = .pred_class)
```

```
##           Truth
## Prediction     0     1
##          0 24898 10739
##          1  6369 19897
```

```
augment(cardio_lda_fit, new_data = cardio_train) %>%
  roc_auc(truth = cardio_disease_1, estimate = .pred_0)
```

```
## # A tibble: 1 x 3
```

```
##     .metric .estimator .estimate
##     <chr>   <chr>          <dbl>
## 1 roc_auc binary          0.789
```

```r
#naive bayes
cardio_nb <- naive_Bayes() %>%
  set_mode('classification') %>%
  set_engine('klaR')

cardio_nb_workflow <- workflow() %>%
  add_model(cardio_nb) %>%
  add_recipe(cardio_recipe)

cardio_nb_fit <- fit(cardio_nb_workflow, cardio_train)

augment(cardio_nb_fit, new_data = cardio_train) %>%
  roc_auc(truth = cardio_disease_1, estimate = .pred_0)
```

```
## # A tibble: 1 x 3
##     .metric .estimator .estimate
##     <chr>   <chr>          <dbl>
## 1 roc_auc binary          0.785
```

```r
#qda
cardio_qda <- discrim_quad() %>%
  set_mode('classification') %>%
  set_engine('MASS')

cardio_qda_workflow <- workflow() %>%
  add_model(cardio_qda) %>%
  add_recipe(cardio_recipe)

cardio_qda_fit <- fit(cardio_qda_workflow, cardio_train)

augment(cardio_qda_fit, new_data = cardio_train) %>%
  roc_auc(truth = cardio_disease_1, estimate = .pred_0)
```

```
## # A tibble: 1 x 3
##     .metric .estimator .estimate
##     <chr>   <chr>          <dbl>
## 1 roc_auc binary          0.756
```

```r
# logistic regression
cardio_log <- logistic_reg() %>%
  set_engine('glm') %>%
  set_mode('classification')

cardio_log_workflow <- workflow() %>%
  add_model(cardio_log) %>%
  add_recipe(cardio_recipe)

cardio_log_fit <- fit(cardio_log_workflow, cardio_train)
```

```
augment(cardio_log_fit, new_data = cardio_train) %>%
  roc_auc(cardio_disease_1, estimate = .pred_0)
```

```
## # A tibble: 1 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 roc_auc binary         0.789
```
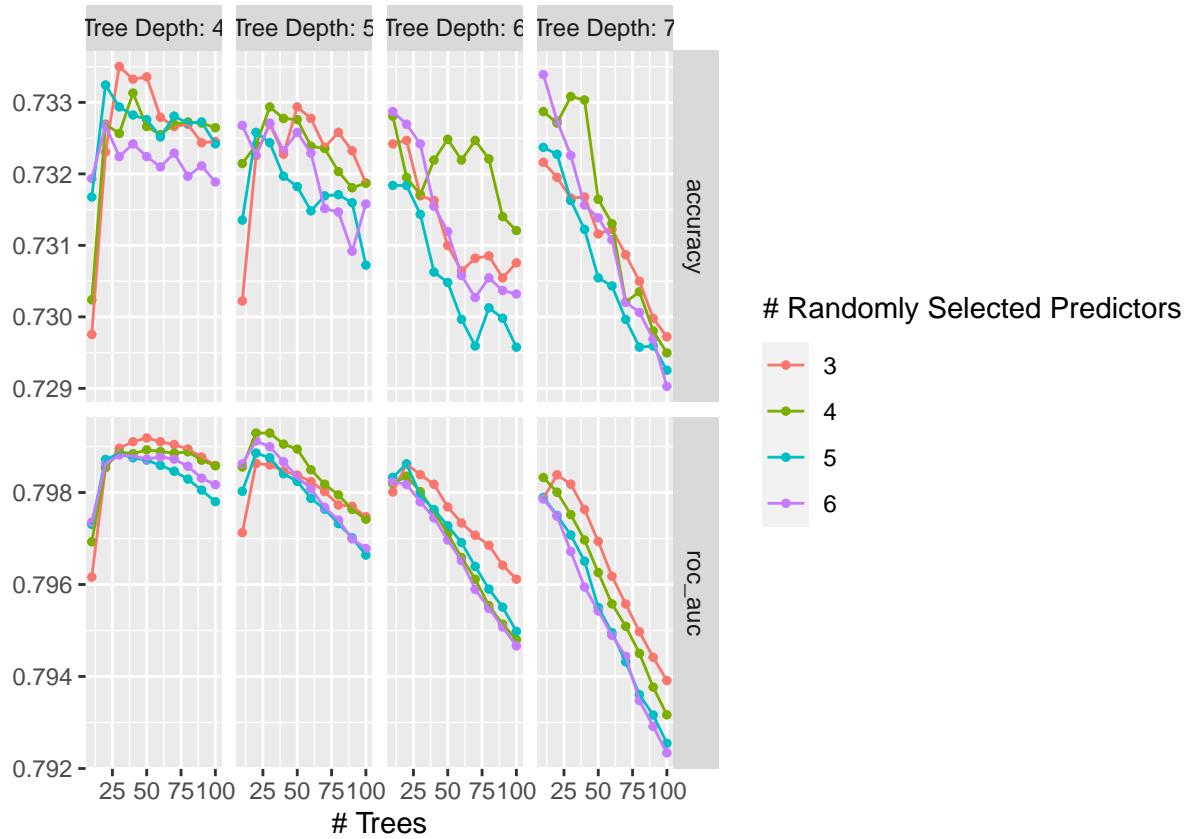
```
load('/Users/andrewchoi/Downloads/PSTAT131_FINAL_PROJECT/PSTAT131_FINAL_PROJECT/Random_Forest_Model.rda
load('/Users/andrewchoi/Downloads/PSTAT131_FINAL_PROJECT/PSTAT131_FINAL_PROJECT/Boosted_Model.rda')
load('/Users/andrewchoi/Downloads/PSTAT131_FINAL_PROJECT/PSTAT131_FINAL_PROJECT/Support_Vector_Machine_
```

```
autoplot(cardio_forest_tune_res)
```
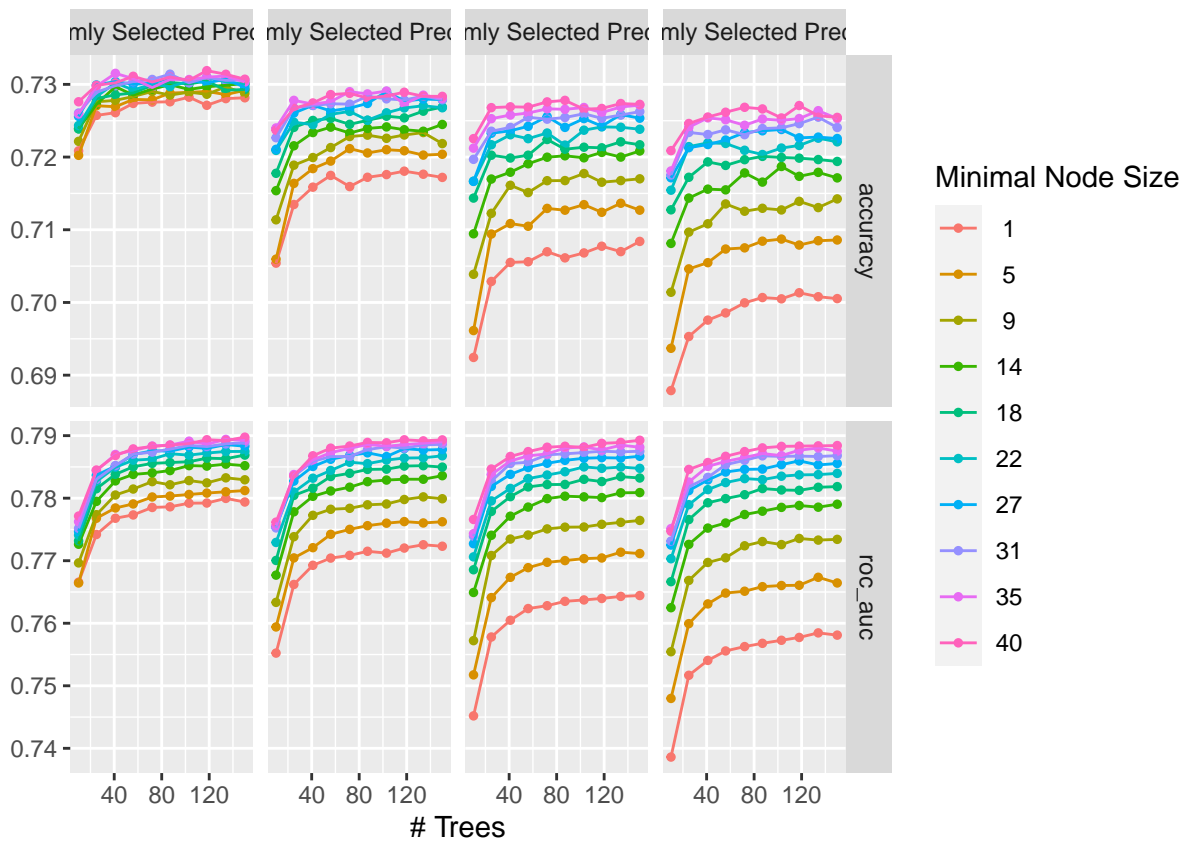


```
autoplot(cardio_boost_tune_res)
```

```
autoplot(cardio_svm_tune_res)
```