PSTAT131_Final Project

Tao Wang

2022-05-24

Contents

Introduction

The purpose of this project

The purpose of this project is to generate a model that will predict whether people will get heart disease or not based on some key indicators.

Some facts you need to know about heart disease

"According to the CDC, heart disease is one of the leading causes of death for people of most races in the US (African Americans, American Indians and Alaska Natives, and white people). About half of all Americans (47%) have at least 1 of 3 key risk factors for heart disease: high blood pressure, high cholesterol, and smoking. Other key indicator include diabetic status, obesity (high BMI), not getting enough physical activity or drinking too much alcohol." (From Kaggle website)

In order to give you a better understanding of heart disease, its harm and the key factors that may cause it, here are a few videos for you:

```
#install.packages('vembedr')
library(vembedr)
embed_url("https://www.youtube.com/watch?v=g131j2lb3xw")
```

```
embed_url("https://www.youtube.com/watch?v=u7k6sqTxOCU&t=139s")
```

Why might this model be useful?

In fact, not only in the United States, heart disease is also an important cause of death in the world. Moreover, the number of deaths from heart disease is increasing every year in the world. Therefore, we need a model to predict people's risk of heart disease according to some important indicators, such as their BMI, whether they smoke, whether they drink alcohol and so on. We also want to find which variables have a significant effect on the likelihood of heart disease. By using this model, we can remind people to pay attention to health and lifestyle habits to help them avoid heart disease.

An overview of my dataset

This project uses Kamil Pytlak's dataset from Kaggle.

According to Kamil Pytlak, his dataset originally comes from the 2020 annual CDC survey data of 400k adults related to their health status. However, he cleaned the original CDC survey data and selected the most relevant variables from it in order to help us to do the machine learning projects related to heart disease.

This dataset contains 319795 observations with 18 variables (9 booleans, 5 strings and 4 decimals). HeartDisease is the response variable. Other 17 variables are our predictors. The full copy of the codebook of this unprocessed dataset available in my zipped files, but in order to help us to better understand these variables,I also show some important parts of this unprocessed dataset's codebook of here.

- `HeartDisease`: Whether the respondent has ever been diagnosed with heart disease
- `BMI`: The body mass index of the respondent
- `Smoking`: Whether the respondent has smoked 100 cigarettes in his/her entire life. [Note: 5 packs = 100 cigarettes]
- `AlcoholDrinking`: Whether the respondent is a heavy drinkers [Note: A heavy drinker is a adult men who having more than 14 drinks per week or a adult women who having more than 7 drinks per week]
- `Stroke`: Whether the respondent had a stroke
- `PhysicalHealth`: The number of days that the respondent had poor physical health in the past 30 days [Note: It includes physical illness and injury]
- `MentalHealth`: The number of days that the respondent had poor mental health in the past 30 days
- `DiffWalking`: Whether the respondent has serious difficulty walking or climbing stairs
- `Sex`: Whether the respondent is male or female
- `AgeCategory`: What age group is the respondent in [Note: the answer should be ‘18-24’, ‘25-29’, ‘30-34’, ‘35-39’, ‘40-44’, ‘45-49’, ‘50-54’, ’55-59’, ’60-64’, ’65-69’, ’70-74’, ’75-79’, ’80 or older’]
- `Race`: The race of the respondent [Note: the answer should be ‘American Indian/Alaskan Native’, ‘Asian’, ‘Black’, ‘Hispanic’, ‘White’, ‘Other’]
- `Diabetes`: Whether the respondent had diabetes [Notes : the answer should be ‘No’, ’No, borderline diabetes’, ‘Yes’, ‘Yes (during pregnancy)’]
- `PhysicalActivity`: Whether the respondent did physical activity or exercise during the past 30 days other than their regular job
- `GenHealth`: The respondent's health assessment of his/her self in general [Notes : the answer should be ‘Very good’, ‘Good’, ‘Excellent’, ‘Fair’, ‘Poor’]
- `SleepTime`: The number of hours of sleep of the respondent in a 24-hour period
- `Asthma`: Whether the respondent had asthma
- `KidneyDisease`: Whether the respondent had kidney disease [Note: not including kidney stones, bladder infection or incontinence]
- `SkinCancer`: Whether the respondent had skin cancer

Note: a full copy of the codebook is available in my zipped files.

Loading Data and Packages

```
# install.packages('caret')
# install.packages("ROSE")
```

```
# load packagges
library(tidyverse)
library(tidymodels)
library(ISLR)
library(ISLR2)
library(discrim)
library(corrr)
library(rpart.plot)
library(vip)
library(janitor)
library(randomForest)
library(xgboost)
library(corrplot)
library(glmnet)
library(ranger)
library(caret)
library(klaR)
library(dplyr)
tidymodels_prefer()
library(ROSE)
set.seed(1234)
```

```
# read the the dataset
records<- read.csv("heart_2020_cleaned.csv")
head(records) # show the first few rows of the dataset
```

```
##   HeartDisease   BMI Smoking AlcoholDrinking Stroke PhysicalHealth MentalHealth
## 1           No 16.60     Yes              No     No              3           30
## 2           No 20.34      No              No    Yes              0            0
## 3           No 26.58     Yes              No     No             20           30
## 4           No 24.21      No              No     No              0            0
## 5           No 23.71      No              No     No             28            0
## 6          Yes 28.87     Yes              No     No              6            0
##   DiffWalking    Sex AgeCategory  Race Diabetic PhysicalActivity GenHealth
## 1          No Female       55-59 White      Yes              Yes Very good
## 2          No Female 80 or older White       No              Yes Very good
## 3          No   Male       65-69 White      Yes              Yes      Fair
## 4          No Female       75-79 White       No               No      Good
## 5         Yes Female       40-44 White       No              Yes Very good
## 6         Yes Female       75-79 Black       No               No      Fair
##   SleepTime Asthma KidneyDisease SkinCancer
## 1         5    Yes            No        Yes
## 2         7     No            No         No
## 3         8    Yes            No         No
## 4         6     No            No        Yes
## 5         8     No            No         No
## 6        12     No            No         No
```

Data Cleaning

While the data set that was downloaded was tidy, a few different cleaning steps were necessary before the split occurred:

- Clean names

```
records <-records %>% clean_names()
head(records)
```

```
##   heart_disease   bmi smoking alcohol_drinking stroke physical_health
## 1            No 16.60     Yes               No     No               3
## 2            No 20.34      No               No    Yes               0
## 3            No 26.58     Yes               No     No              20
## 4            No 24.21      No               No     No               0
## 5            No 23.71      No               No     No              28
## 6           Yes 28.87     Yes               No     No               6
##   mental_health diff_walking    sex age_category  race diabetic
## 1            30           No Female        55-59 White      Yes
## 2             0           No Female  80 or older White       No
## 3            30           No   Male        65-69 White      Yes
## 4             0           No Female        75-79 White       No
## 5             0          Yes Female        40-44 White       No
## 6             0          Yes Female        75-79 Black       No
##   physical_activity gen_health sleep_time asthma kidney_disease skin_cancer
## 1               Yes  Very good          5    Yes             No         Yes
## 2               Yes  Very good          7     No             No          No
## 3               Yes       Fair          8    Yes             No          No
## 4                No       Good          6     No             No         Yes
## 5               Yes  Very good          8     No             No          No
## 6                No       Fair         12     No             No          No
```

- Deal with imbalanced problems Now, let check whether our response variable is balanced or not. If not, we need to deal with it.

```
table(records$heart_disease)
```

```
##
##     No    Yes
## 292422  27373
```

We find that our response variable is highly imbalanced. There are much more observations on 'No' levels than 'Yes' levels. We need to use some functions to deal with this problem, otherwise it will have a serious impact on our predictions. According to the TA, we can use ovun.sample() function to help us to deal with it.

```
set.seed(1234)
newrecords<- ovun.sample(heart_disease~.,data = records,
                             p=0.5,seed = 1,method = "under")$data
```

- Check whether our response variable are balanced or not.

```
table(newrecords$heart_disease)
```

```
##
##    No   Yes
## 27387 27373
```

Although there are more 'No', our response variable is almost balanced.

- Check if there is a missing value. If yes, we need to remove the observation with the missing value. If no, we continue the process of data cleaning.

```
sum(is.na(newrecords))
```

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

```
# It means there is no missing value in our data.
# We continue the process of data cleaning.
```

- Convert heart_disease, smoking, alcohol_drinking, stroke, diff_walking, sex, age_category, race, diabetic, physical_activity, gen_health, asthma, kidney_disease, skin_cancer to factors

```
newrecords <- newrecords %>%
  mutate(
    heart_disease  = factor(heart_disease, levels = c('Yes', 'No')),
    smoking  = factor(smoking, levels = c('Yes', 'No')),
    alcohol_drinking  = factor(alcohol_drinking, levels = c('Yes', 'No')),
    stroke  = factor(stroke, levels = c('Yes', 'No')),
    diff_walking  = factor(diff_walking, levels = c('Yes', 'No')),
    sex  = factor(sex),
    age_category  = factor(age_category),
    race  = factor(race, levels = c("American Indian/Alaskan Native", "Asian", "Black", "Hispanic", "Whi
    diabetic  = factor(diabetic, levels = c("No", "No, borderline diabetes", "Yes", "Yes (during pregnan
    physical_activity  = factor(physical_activity),
    gen_health  = factor(gen_health, levels = c("Excellent","Very good", "Good", "Fair", "Poor" )),
    asthma  = factor(asthma, levels = c('Yes', 'No')),
    kidney_disease  = factor(kidney_disease, levels = c('Yes', 'No')),
    skin_cancer  = factor(skin_cancer, levels = c('Yes', 'No')),
  )
head(newrecords)
```

```
##   heart_disease   bmi smoking alcohol_drinking stroke physical_health
## 1            No 33.84      No               No     No               0
## 2            No 31.75      No              Yes     No               0
## 3            No 33.64      No               No     No               0
## 4            No 24.56      No               No     No               0
## 5            No 40.69     Yes               No     No              30
## 6            No 27.89      No               No     No               0
##   mental_health diff_walking    sex age_category  race diabetic
## 1             2           No Female        45-49 White       No
## 2             0           No   Male        55-59 White       No
## 3            28           No   Male        40-44 Black      Yes
## 4             0           No Female        40-44 Asian       No
## 5             0          Yes   Male        60-64 White      Yes
## 6             0           No   Male        75-79 White       No
##   physical_activity gen_health sleep_time asthma kidney_disease skin_cancer
## 1               Yes  Very good          6     No             No         Yes
## 2               Yes  Excellent          7     No             No         Yes
## 3               Yes       Good          7     No             No          No
## 4               Yes  Excellent          6     No             No          No
## 5               Yes       Fair          7     No             No          No
## 6               Yes  Very good          5     No             No         Yes
```

```
# show me how many observations in the new dataset
# show me how many variables in the new dataset
dim(newrecords)
```

```
## [1] 54760    18
```

- We completed the the process of data cleaning.

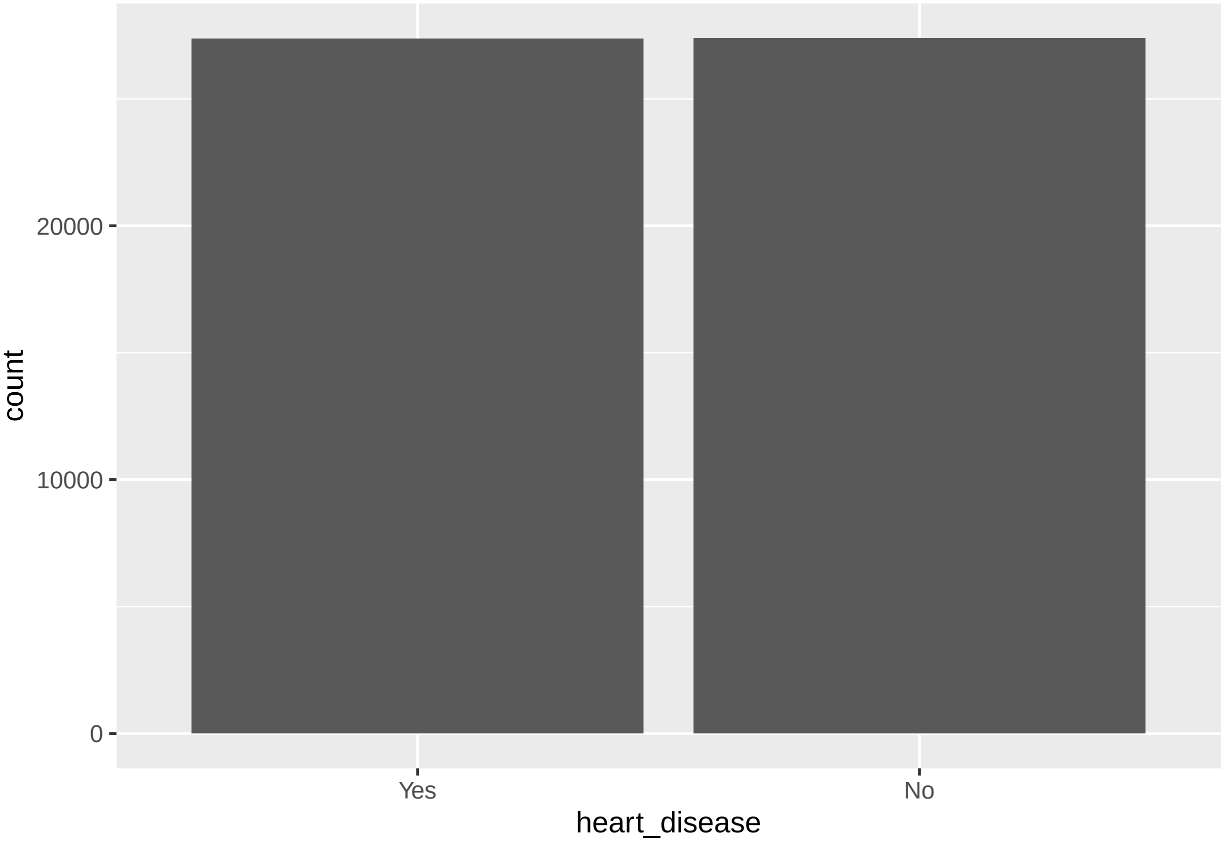
Exploratory Data Analysis

This entire exploratory data analysis will be based only on the entire set, which has 54760 observations with 18 variables. Each observation represents a single `newrecords` class.

Variable heart_disease

- During the process of exploratory data analysis, we first analyze our response variable `heart_disease`.

```
newrecords %>%
  ggplot(aes(x = heart_disease)) +
  geom_bar()
```



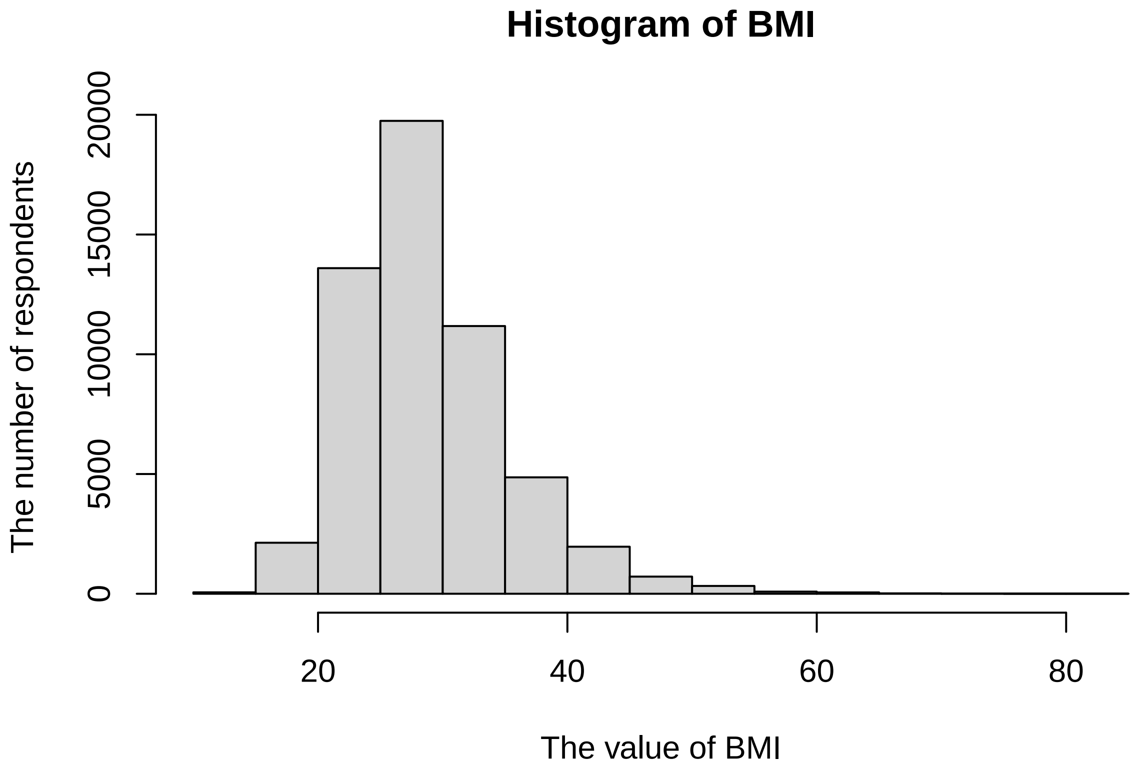
- According to the graph, we find that our response variable is almost balanced. Although there are much more observations on 'No' levels than 'Yes' levels, this will not have a significant impact on our predictions.

Variable bmi

Now, let's analyze variable `bmi`.

- First, draw a histogram of variable `bmi`.

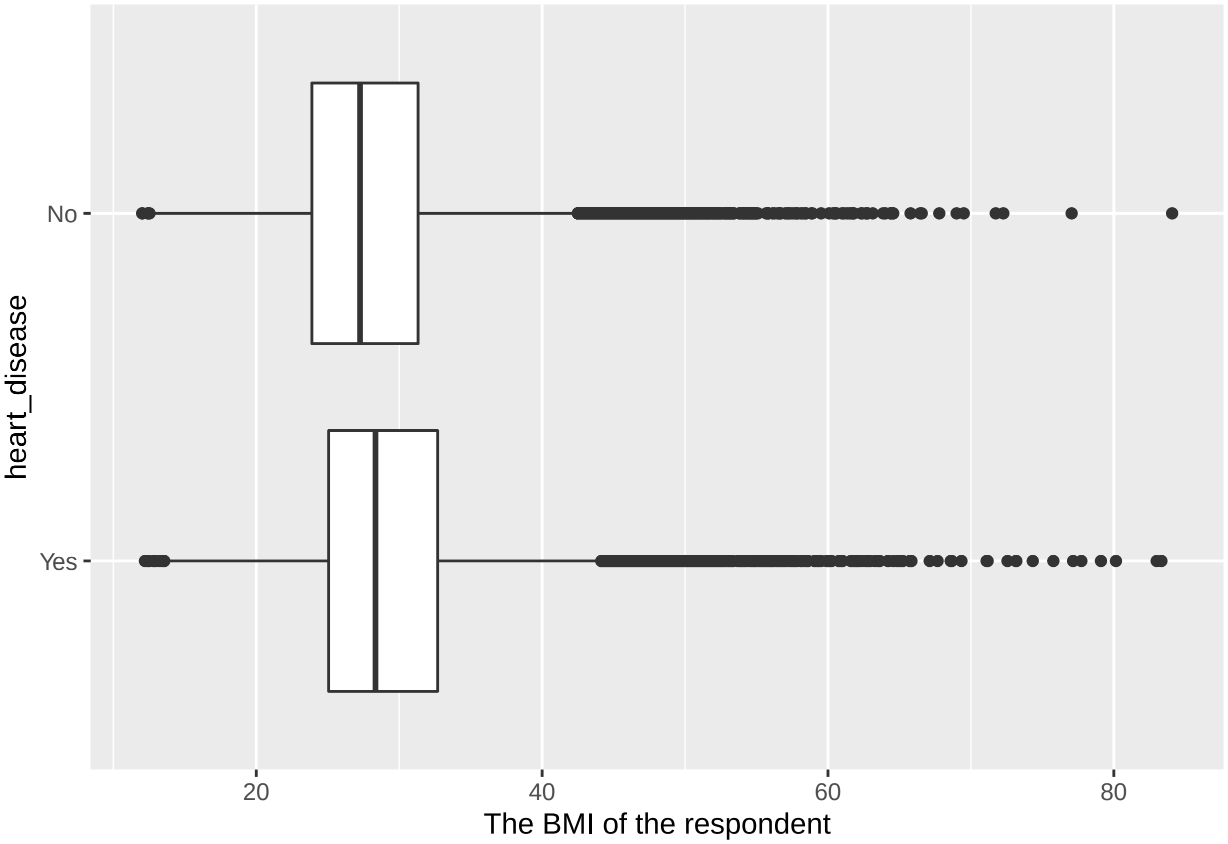
```
hist(newrecords$bmi, main = paste("Histogram of BMI"), xlab = 'The value of BMI', ylab = 'The number of
```



* The distribution of `bmi` definitely appears to be left skewed, and it has a long right tail. It also almost looks a normal distribution. There's one peak around 25-30. Most people have a BMI below 40.

- Second, draw a boxplot of variable `bmi` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x = bmi, y=heart_disease))+
  geom_boxplot() +
  xlab("The BMI of the respondent")
```



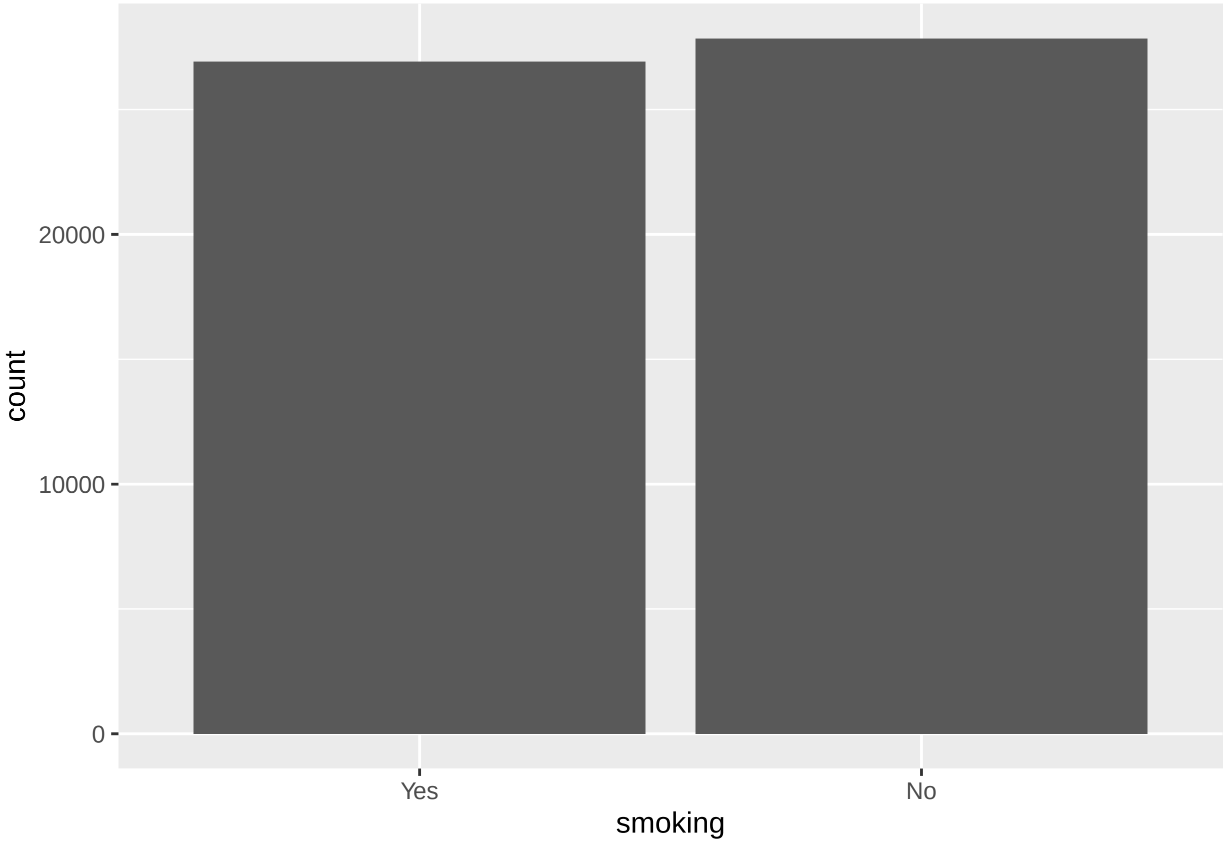
*

Based on the graph, we can find that the respondent who have higher BMI is more likely to get heart disease. (We will confirm this result at the end of this project)

Variable smoking

- First, draw a plot of variable `smoking`.

```
newrecords %>%
  ggplot(aes(x = smoking)) +
  geom_bar()
```

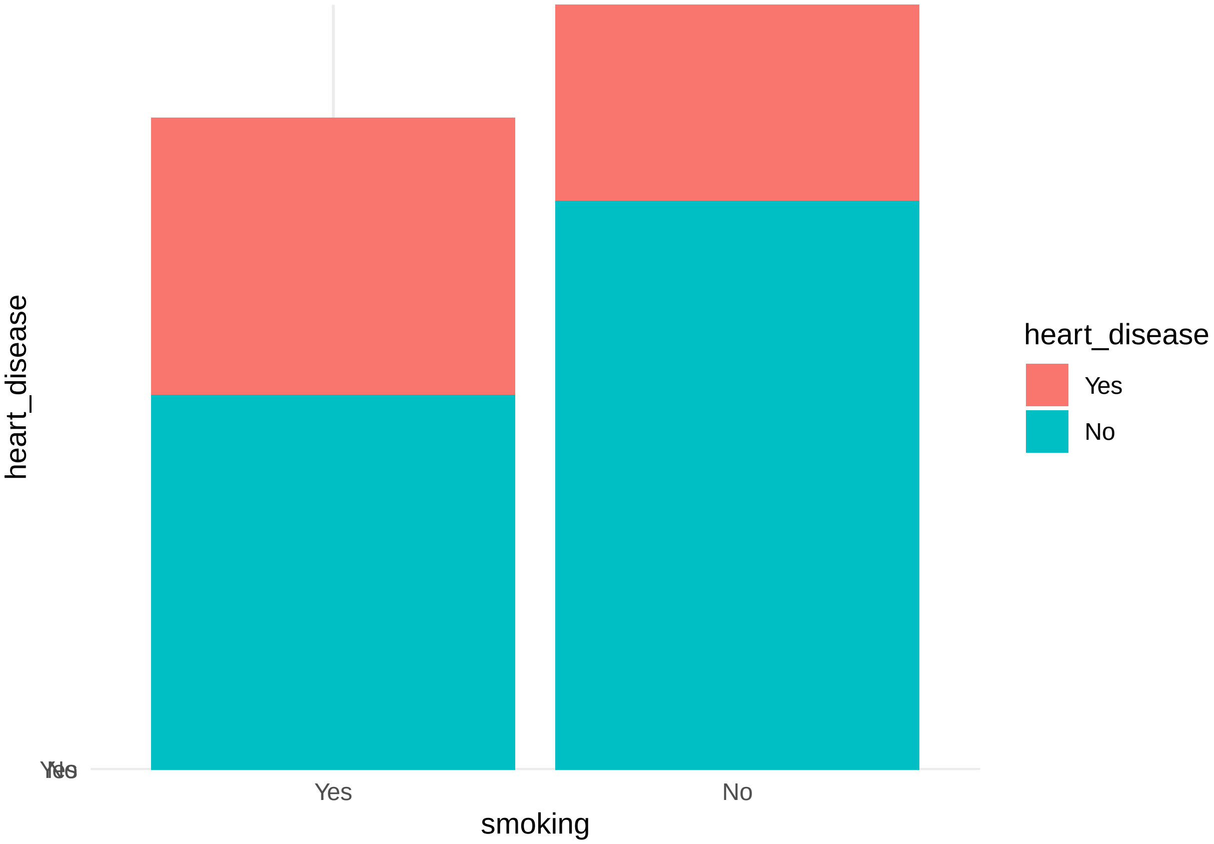


*

According to the graph, we find that there are much more observations on 'No' levels than 'Yes' levels for variable smoking.

- Second, draw a plot of variable `smoking` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= smoking, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```

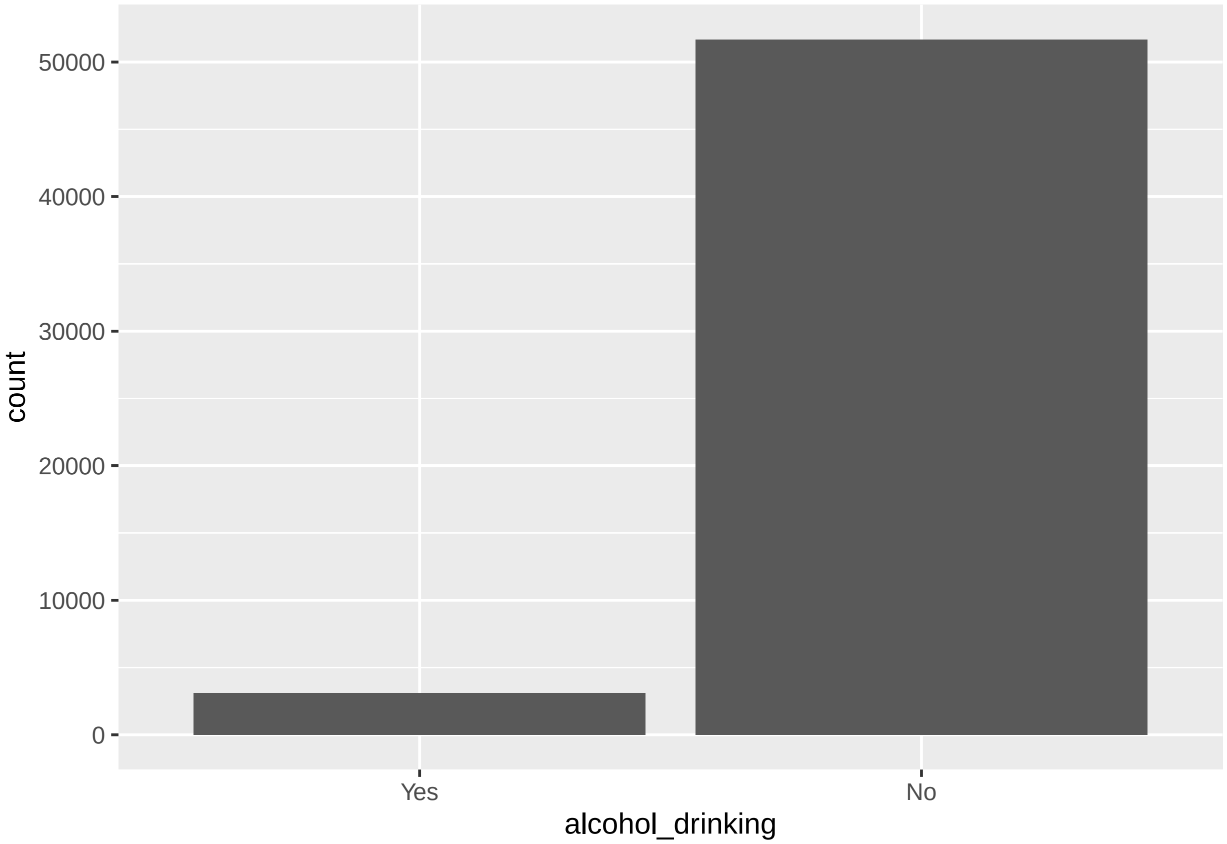


- Based on the graph, we can find that the respondent who likes smoking (has smoked 100 cigarettes in his/her entire life) is more likely to get heart disease. (We will confirm this result at the end of this project)

Variable alcohol_drinking

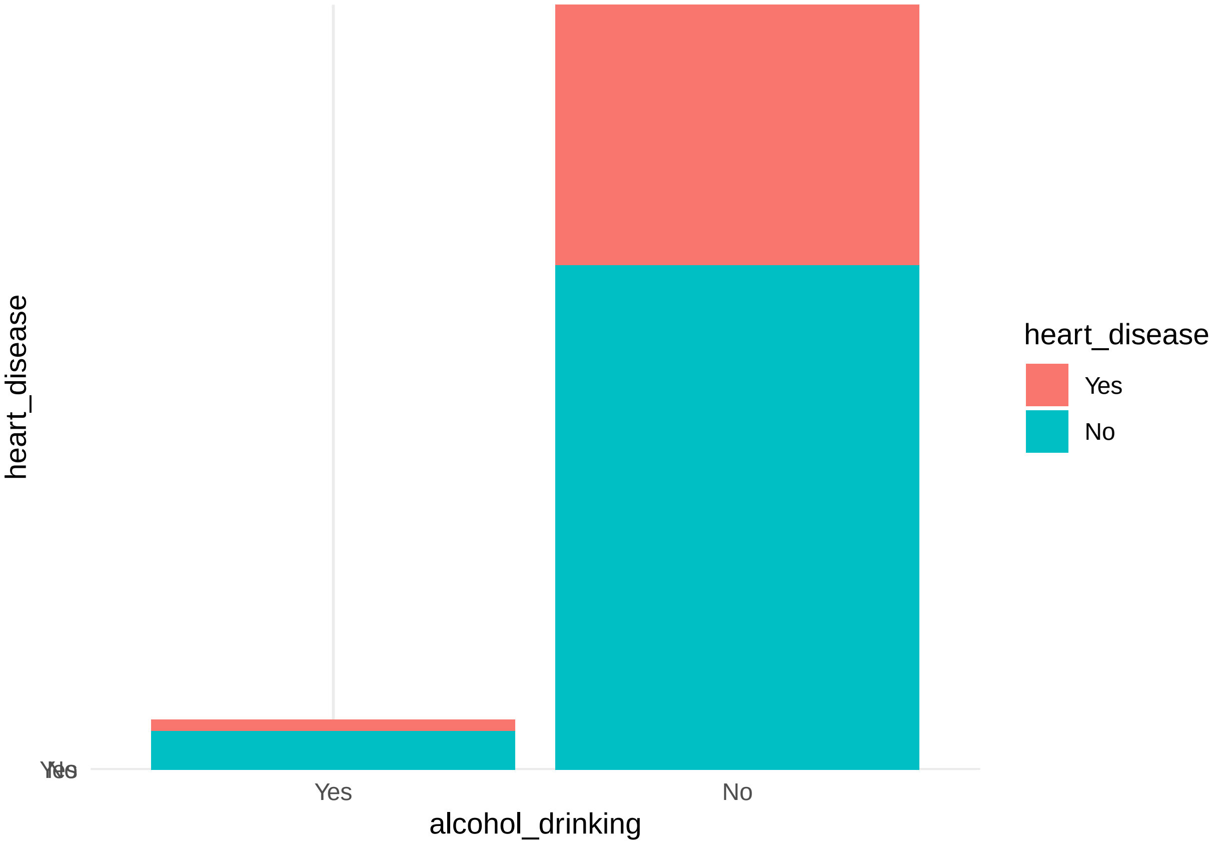
- First, draw a plot of variable `alcohol_drinking`.

```
newrecords %>%
  ggplot(aes(x = alcohol_drinking)) +
  geom_bar()
```



- According to the graph, we find that there are much more observations on ‘No’ levels than ‘Yes’ levels for variable alcohol_drinking. For this reason, it maybe hard for us to find the relationship between `alcohol_drinking` and `heart_disease`.
- Second, draw a plot of variable `alcohol_drinking` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= alcohol_drinking, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```

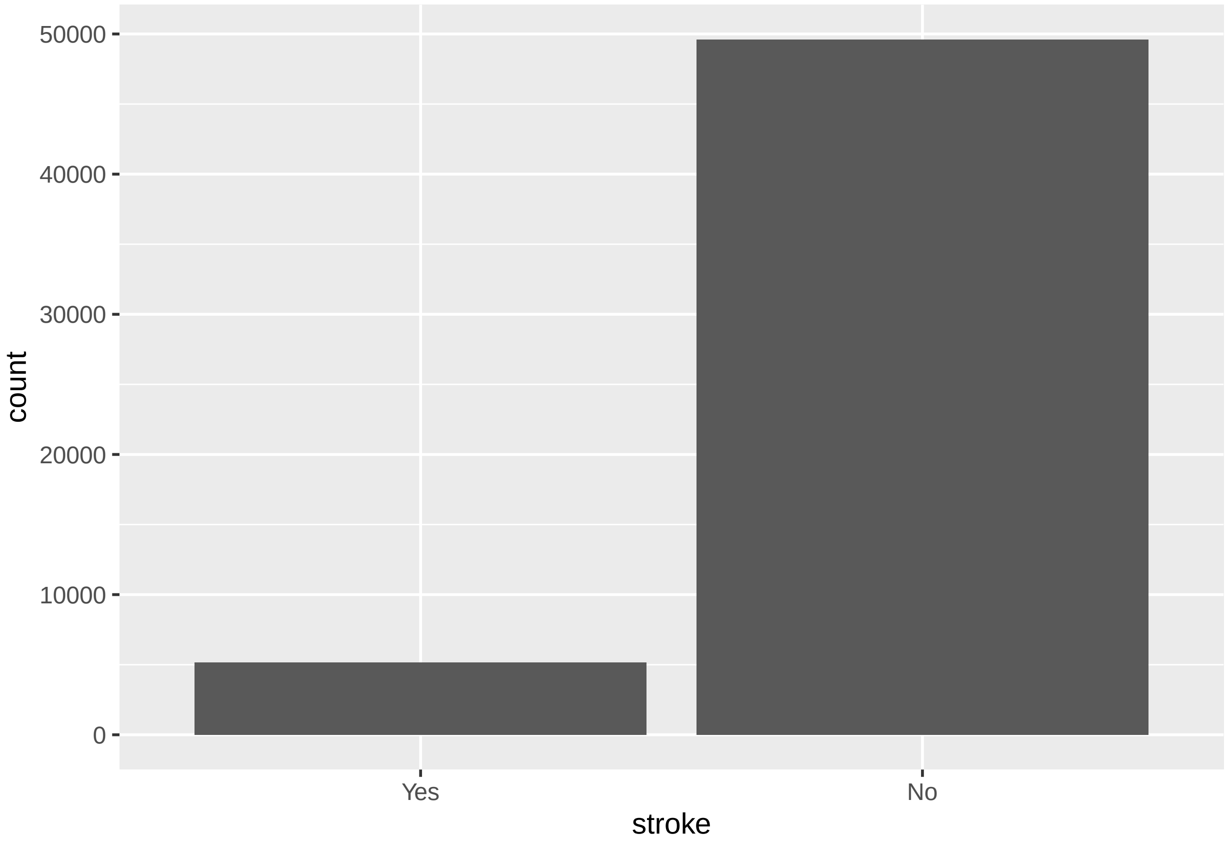


- Based on the graph, we can find that the respondent who is not heavy drinker is more likely to get heart disease. [A heavy drinker is a adult men who having more than 14 drinks per week or a adult women who having more than 7 drinks per week]
- Notice that the result may not be correct since we have much more observations on 'No' levels than 'Yes' levels for variable alcohol_drinking. (We will confirm this result at the end of this project)

Variable Stroke

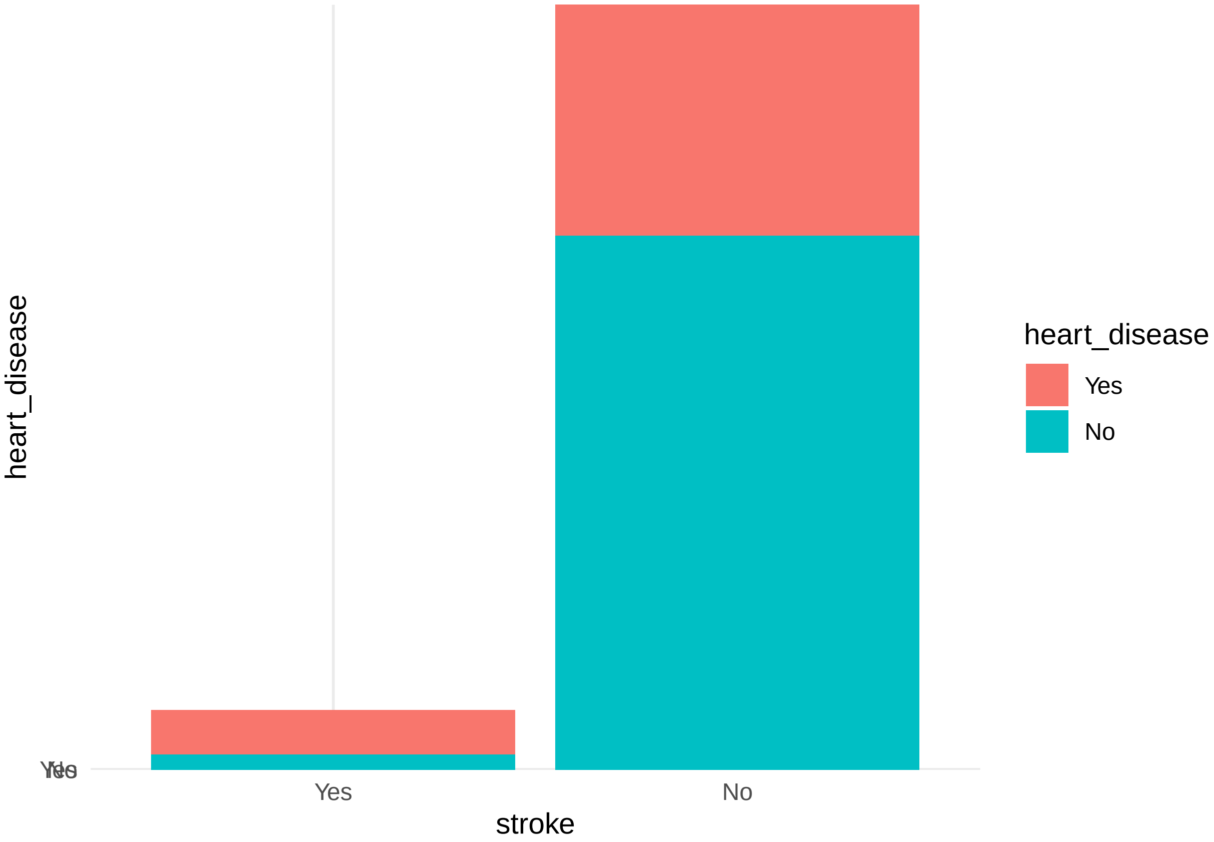
- First, draw a plot of variable `stroke`.

```
newrecords %>%
  ggplot(aes(x = stroke)) +
  geom_bar()
```



- According to the graph, we find that there are much more observations on ‘No’ levels than ‘Yes’ levels for variable `stroke`.For this reason, it maybe hard for us to find the relationship between `stroke` and `heart_disease`.
- Second, draw a plot of variable `stroke` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= stroke, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```

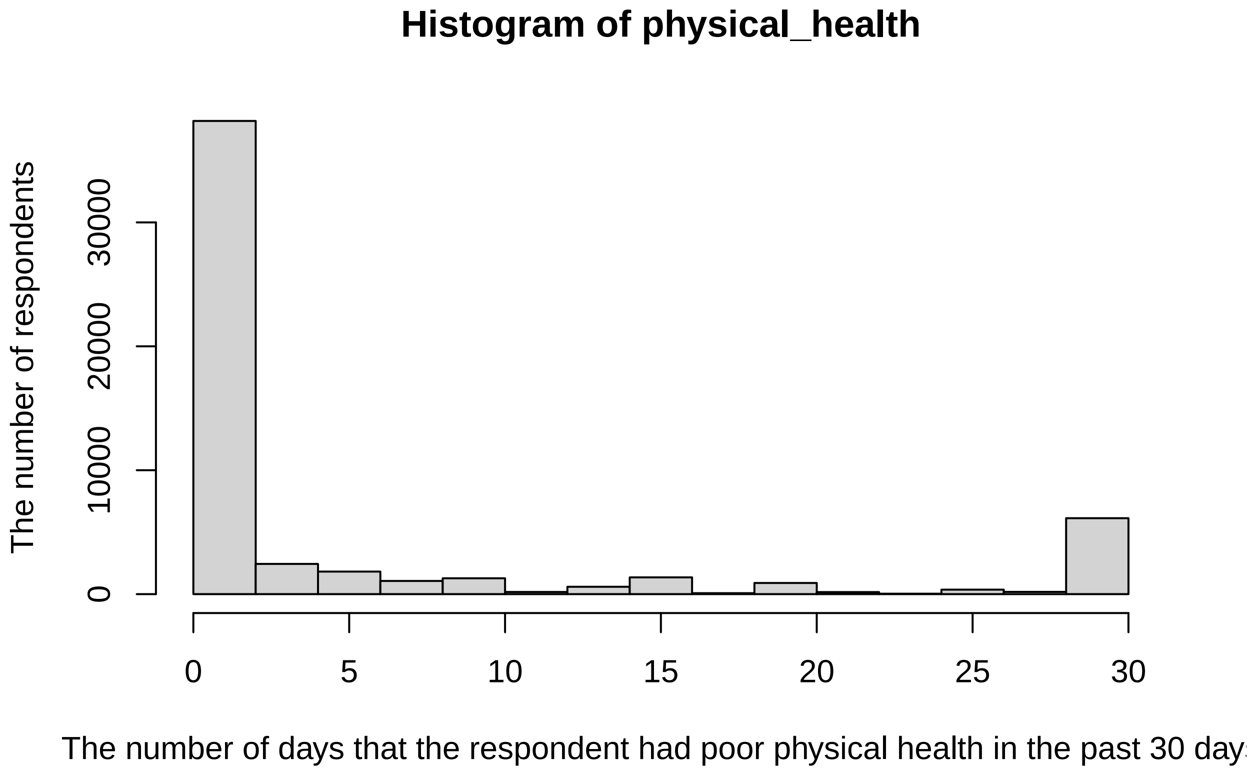


- Based on the graph, we can find that the respondent who had stoke is more likely to get heart disease.(We will confirm this result at the end of this project)

Variable physical_health

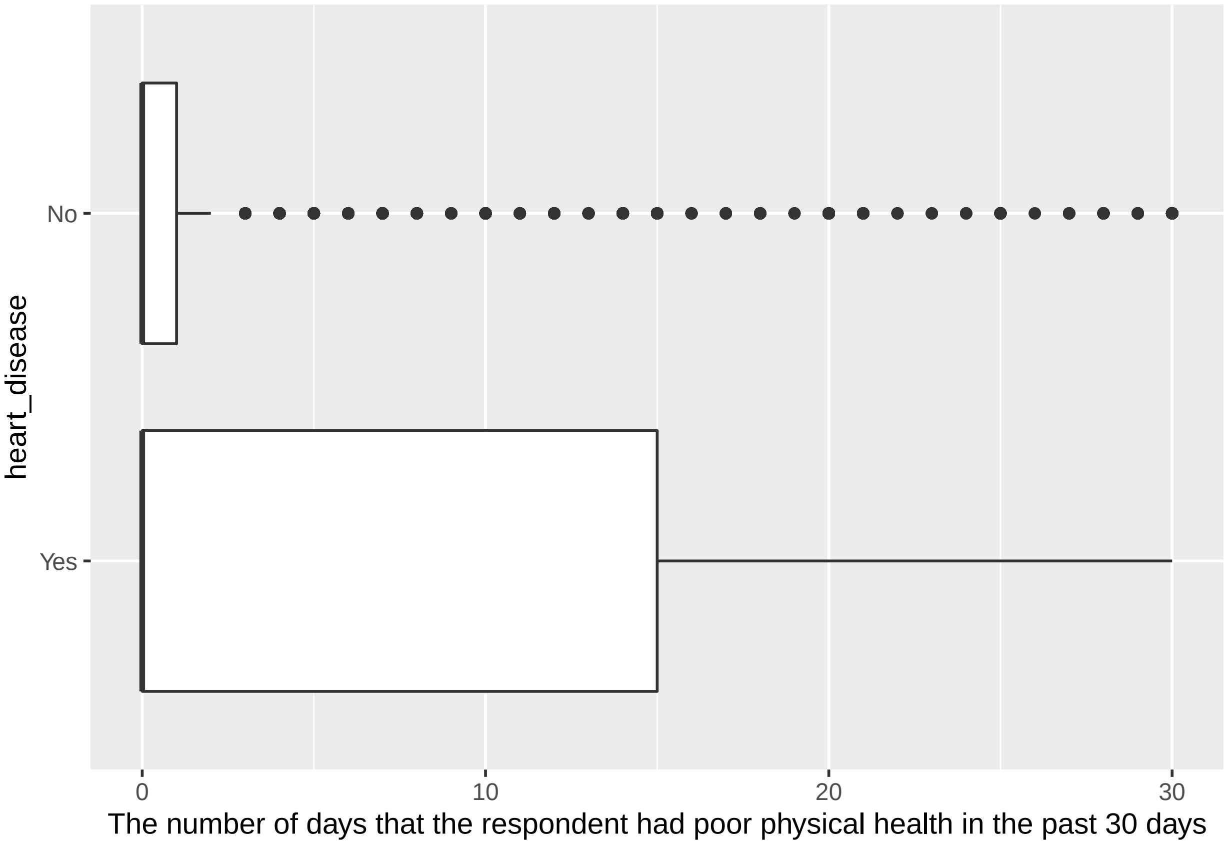
- First, draw a histogram of variable `physical_health`: The number of days that the respondent had poor physical health in the past 30 days [Note: It includes physical illness and injury]

```
hist(newrecords$physical_health, main = paste("Histogram of physical_health"), xlab = 'The number of da
```



- According to the graph, we find that most of respondents had less than 10 poor physical health day in the past 30 days, and there are some respondents had 30 poor physical health day in the past 30 days.
- Second, draw a boxplot of variable `physical_health` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x =physical_health , y=heart_disease))+
  geom_boxplot() +
  xlab("The number of days that the respondent had poor physical health in the past 30 days")
```

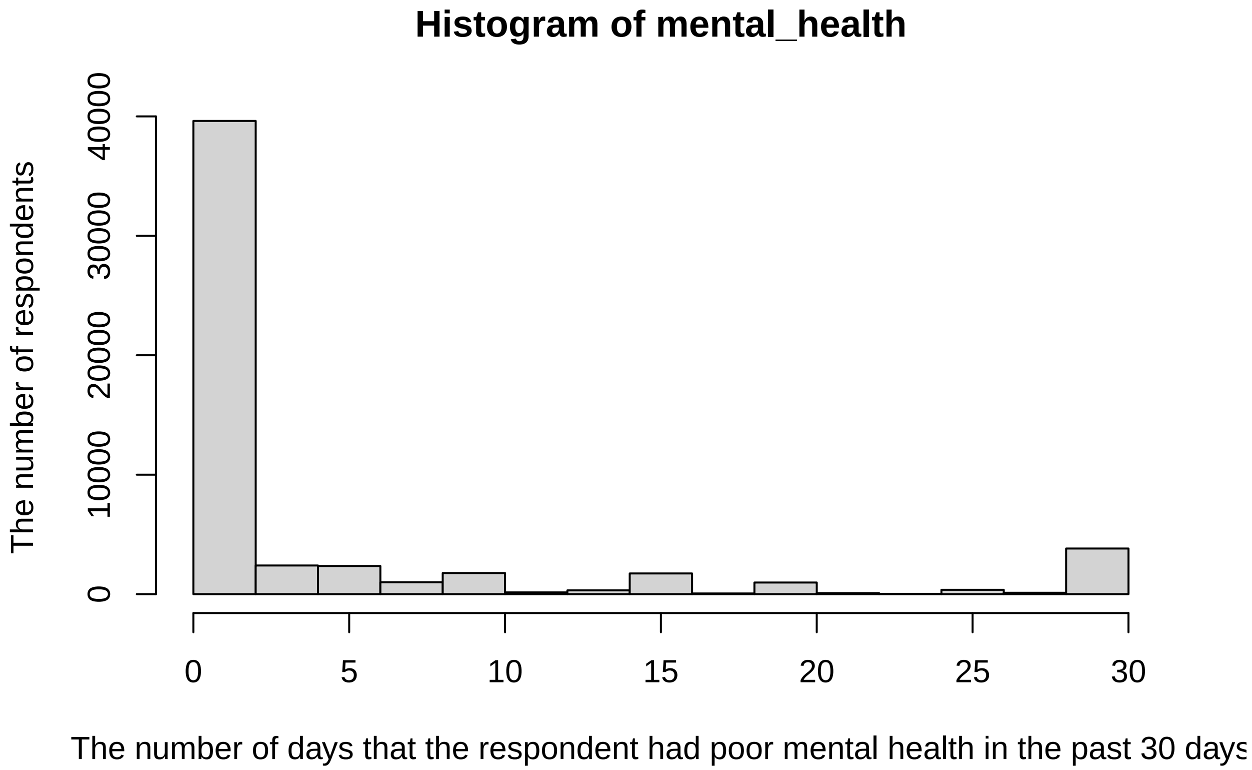


- Based on the graph, we can find that the number of days that the respondents had poor physical health in the past 30 days may affect whether they have a heart disease. (We will confirm this result at the end of this project)

Variable mental_health

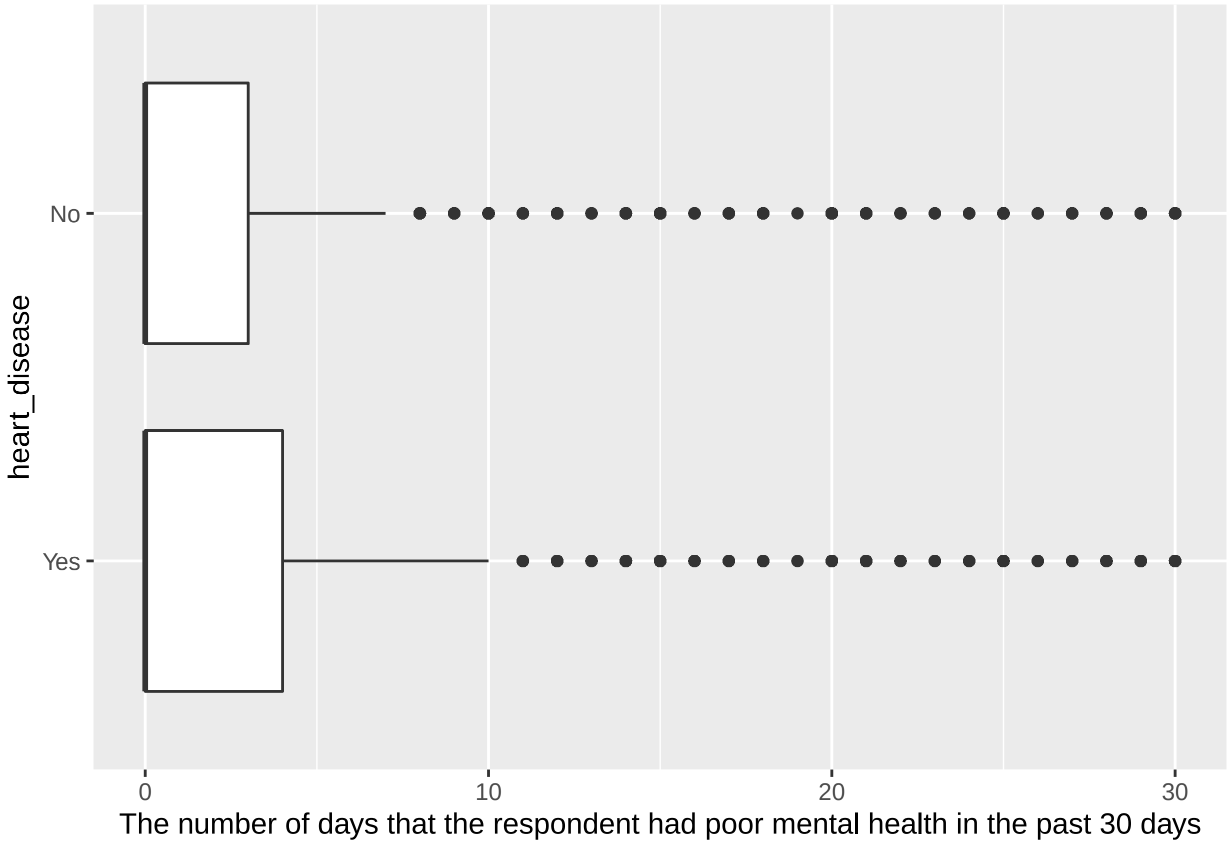
- First, draw a histogram of variable `mental_health`: The number of days that the respondent had poor mental health in the past 30 days

```
hist(newrecords$mental_health, main = paste("Histogram of mental_health"), xlab = 'The number of days th
```



- According to the graph, we find that most of respondents had less than 10 poor mental health day in the past 30 days, and there are some respondents had 30 poor mental health days in the past 30 days.
- Second, draw a boxplot of variable `mental_health` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x =mental_health , y=heart_disease))+
  geom_boxplot() +
  xlab("The number of days that the respondent had poor mental health in the past 30 days")
```

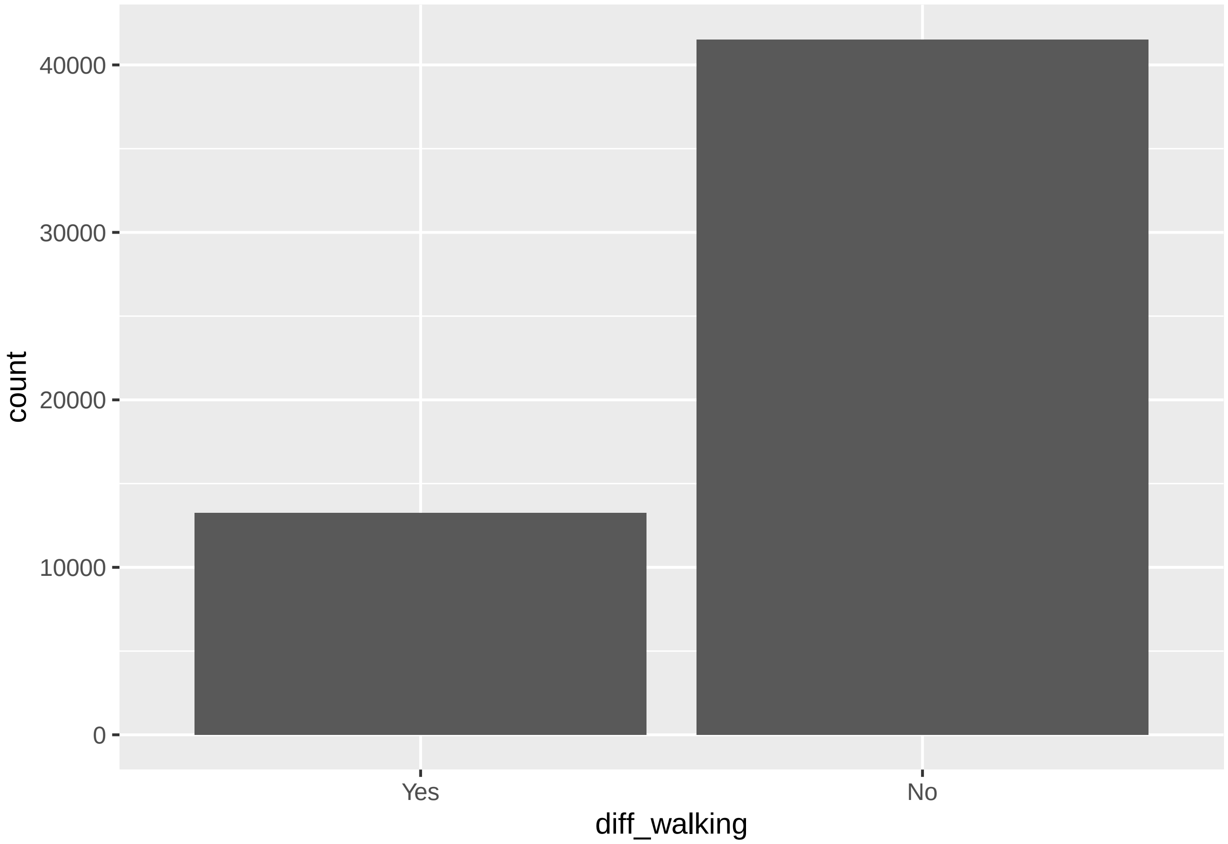


- Based on the graph, we can find that the number of days that the respondents had poor mental health in the past 30 days may affect whether they have a heart disease. (We will confirm this result at the end of this project)

Variable diff_walking

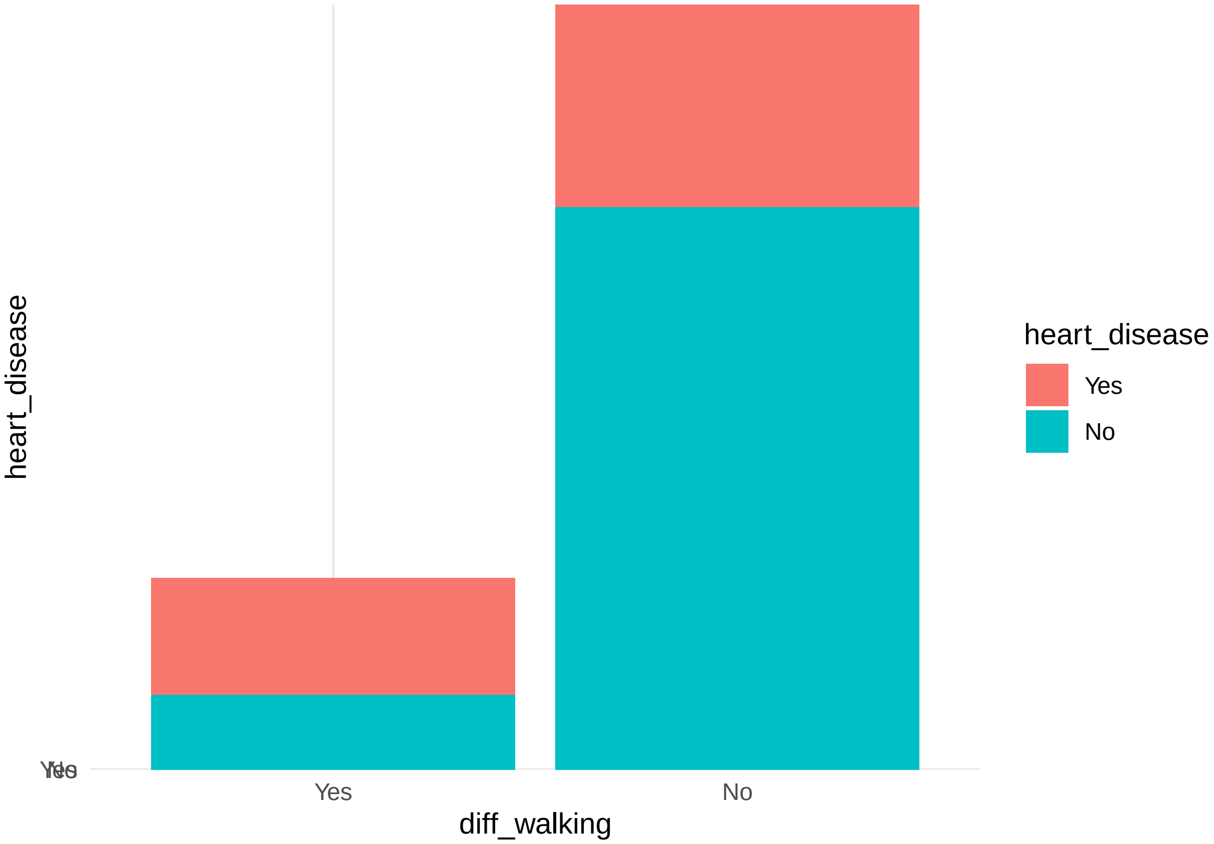
- First, draw a plot of variable `diff_walking`.

```
newrecords %>%
  ggplot(aes(x = diff_walking)) +
  geom_bar()
```



- According to the graph, we find that there are much more observations on 'No' levels than 'Yes' levels for variable `diff_walking`.For this reason, it maybe hard for us to find the relationship between `diff_walking` and `heart_disease`.
- Second, draw a plot of variable `diff_walking` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= diff_walking, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```

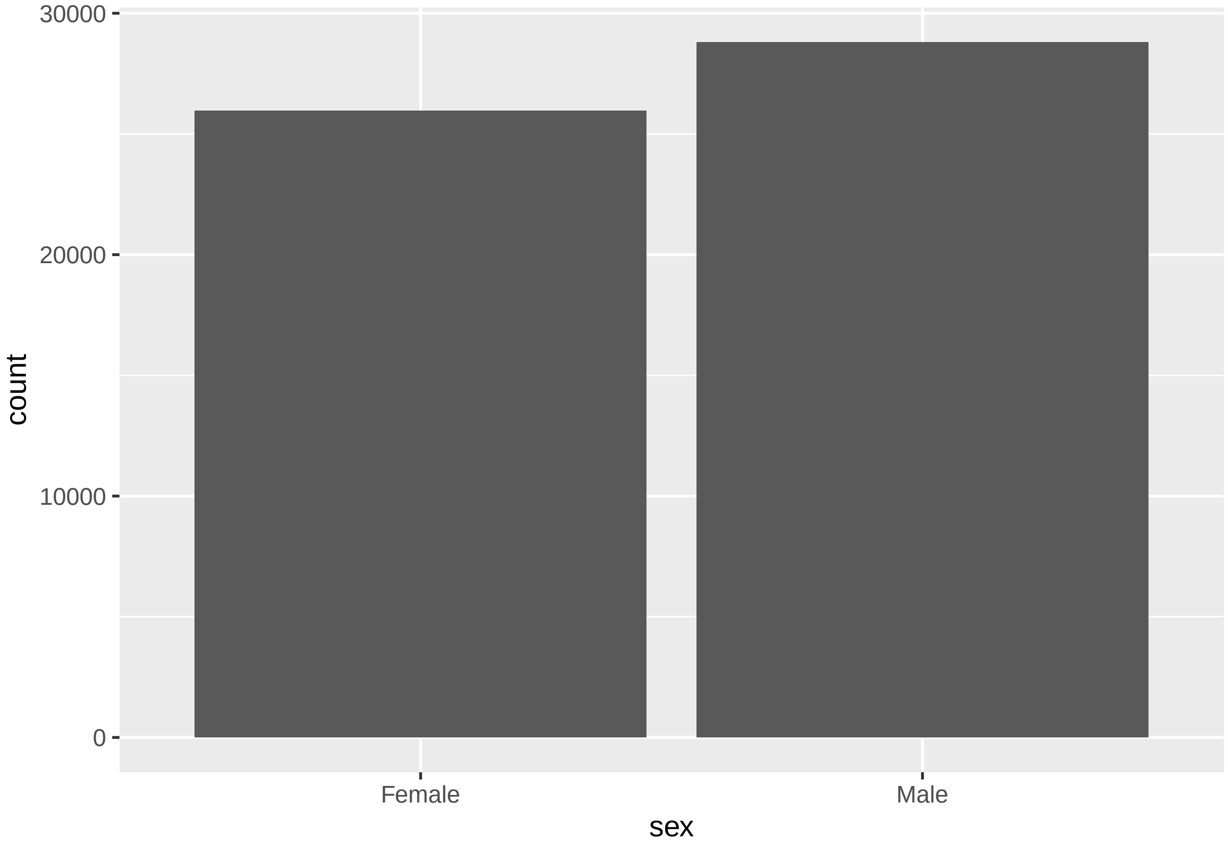


- Based on the graph, we can find that the respondent who has serious difficulty walking or climbing stairs is more likely to get heart disease. (We will confirm this result at the end of this project)

Variable sex

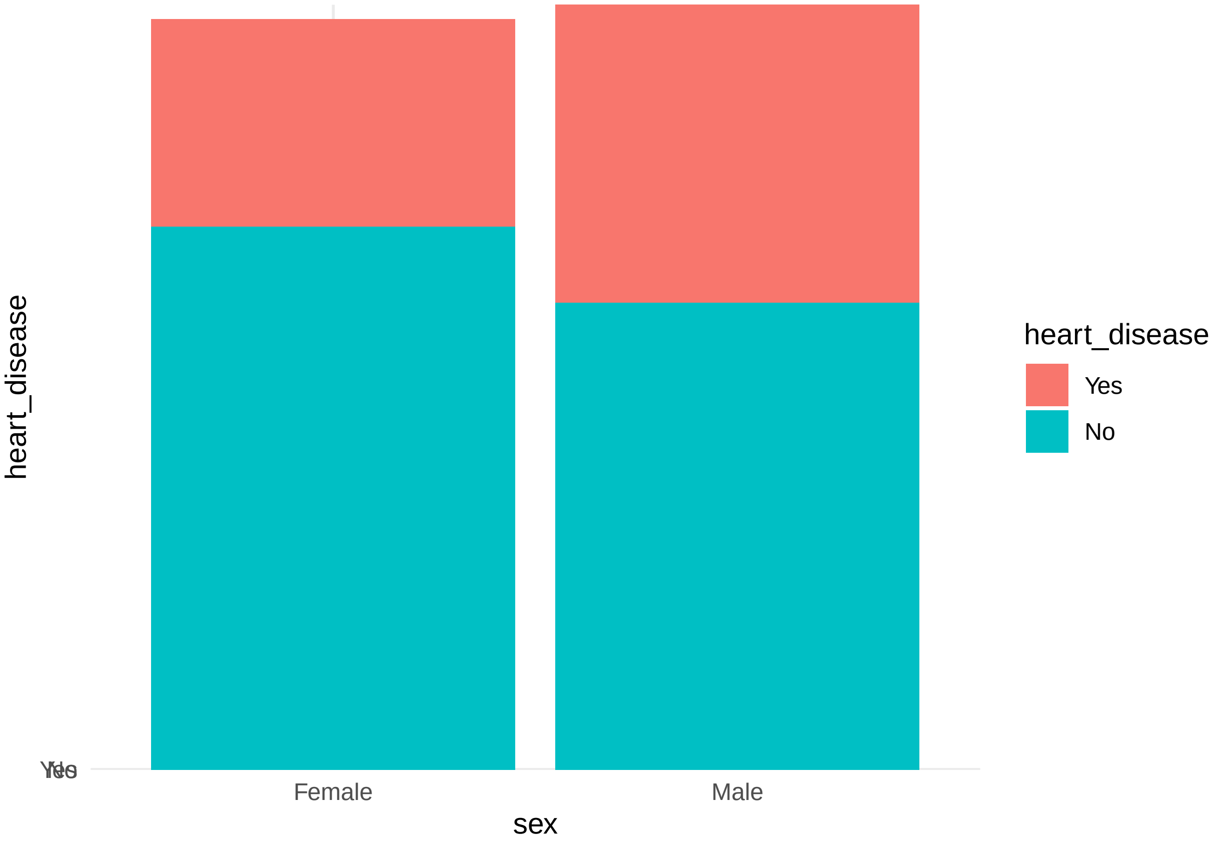
- First, draw a plot of variable `sex`.

```
newrecords %>%
  ggplot(aes(x = sex)) +
  geom_bar()
```



- According to the graph, we find that there are more observations on 'Male' levels than 'Female' levels for variable `sex`. Since the difference between these two levels is no very big, it won't make bad influence.
- Second, draw a plot of variable `sex` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= sex, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```

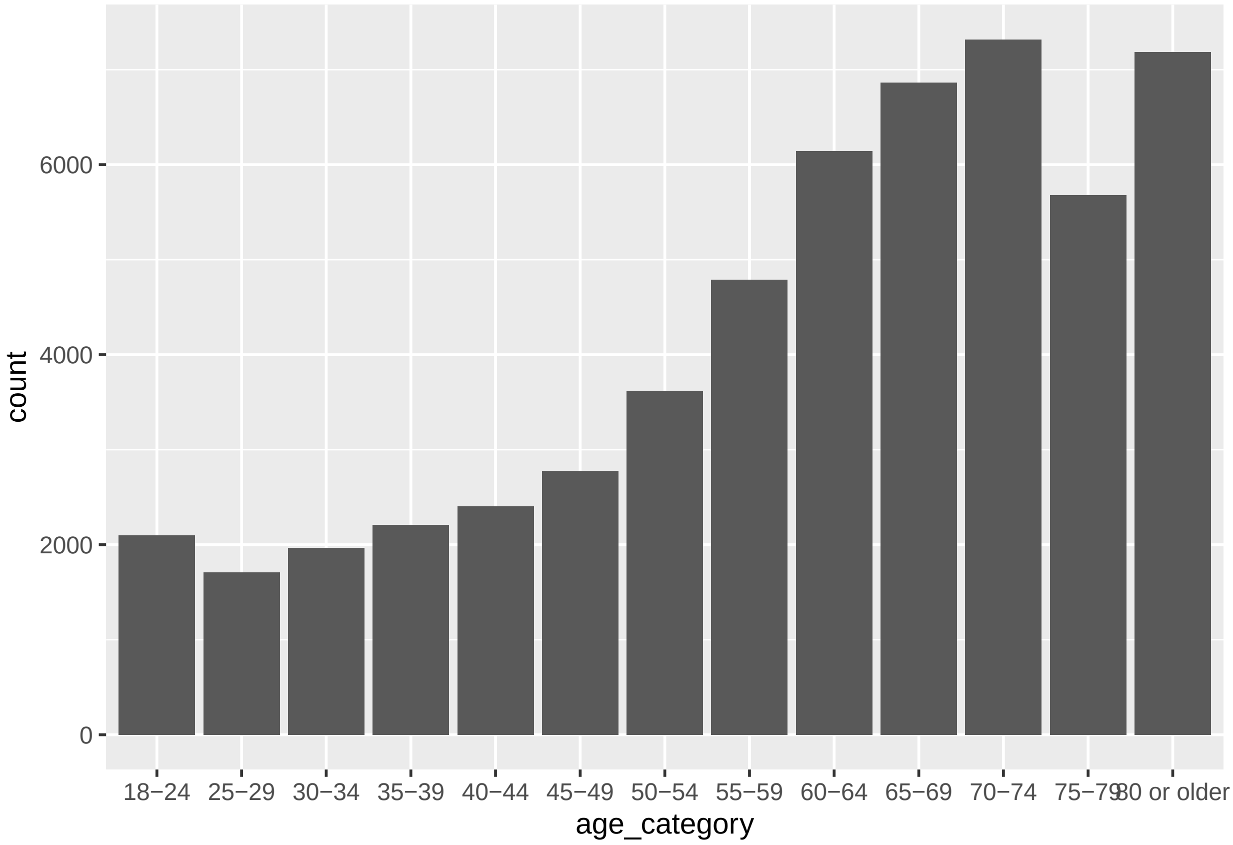


- Based on the graph, we can find that the respondent who is male is more likely to get heart disease. (We will confirm this result at the end of this project)

Variable age_category

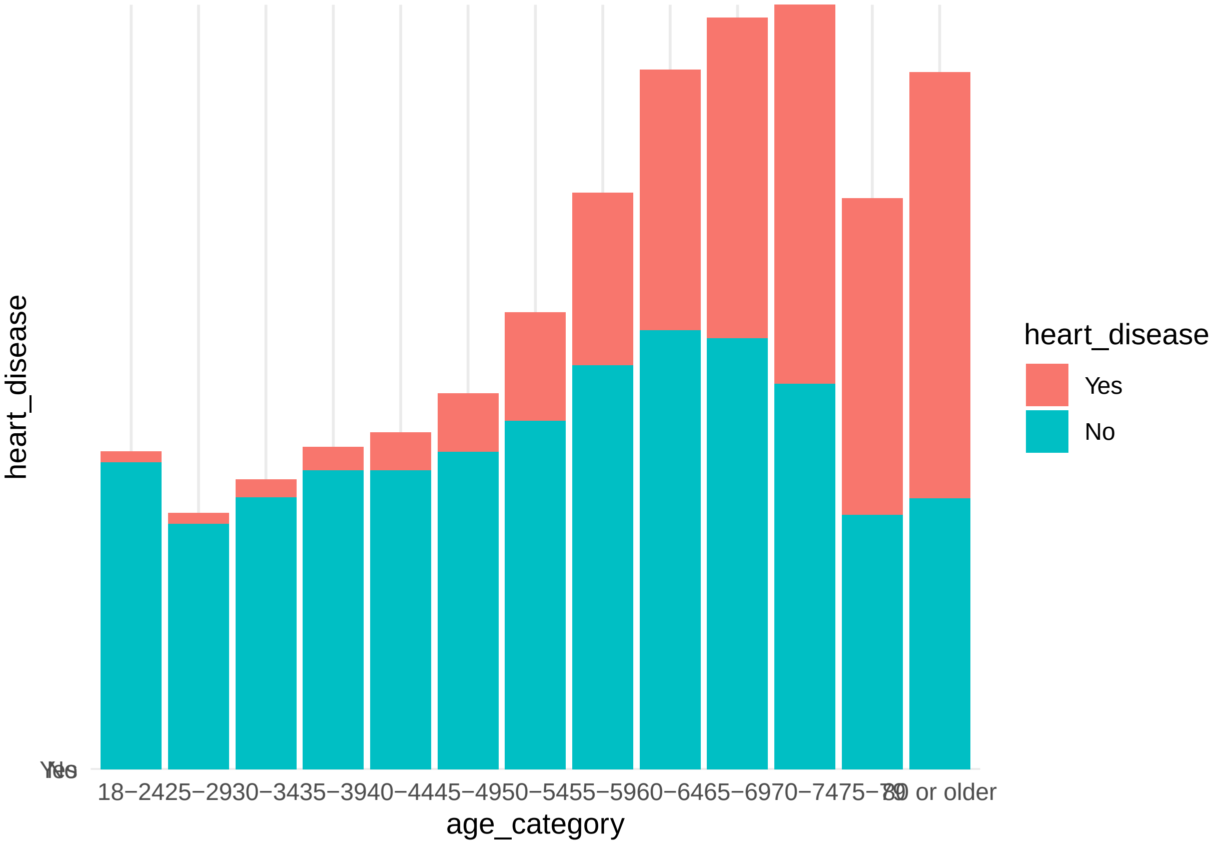
- First, draw a plot of variable `age_category`.

```
newrecords %>%
  ggplot(aes(x = age_category)) +
  geom_bar()
```



- According to the graph, we find that there are more older respondents (older than 50) than young respondents (younger than 50).
- Second, draw a plot of variable `age_category` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= age_category, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```

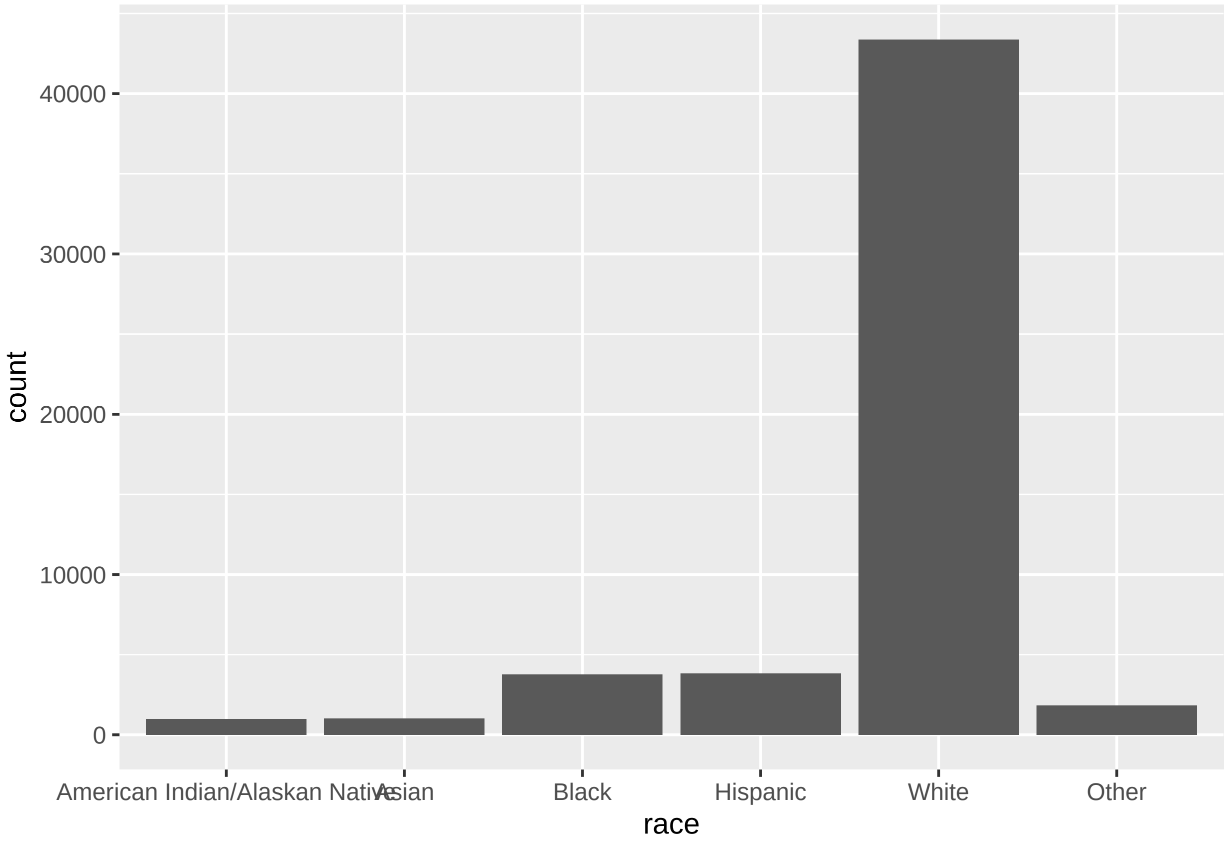


- Based on the graph, we can find that older people (older than 50) are more likely to get heart disease than younger people (less than 50). (We will confirm this result at the end of this project)

Variable race

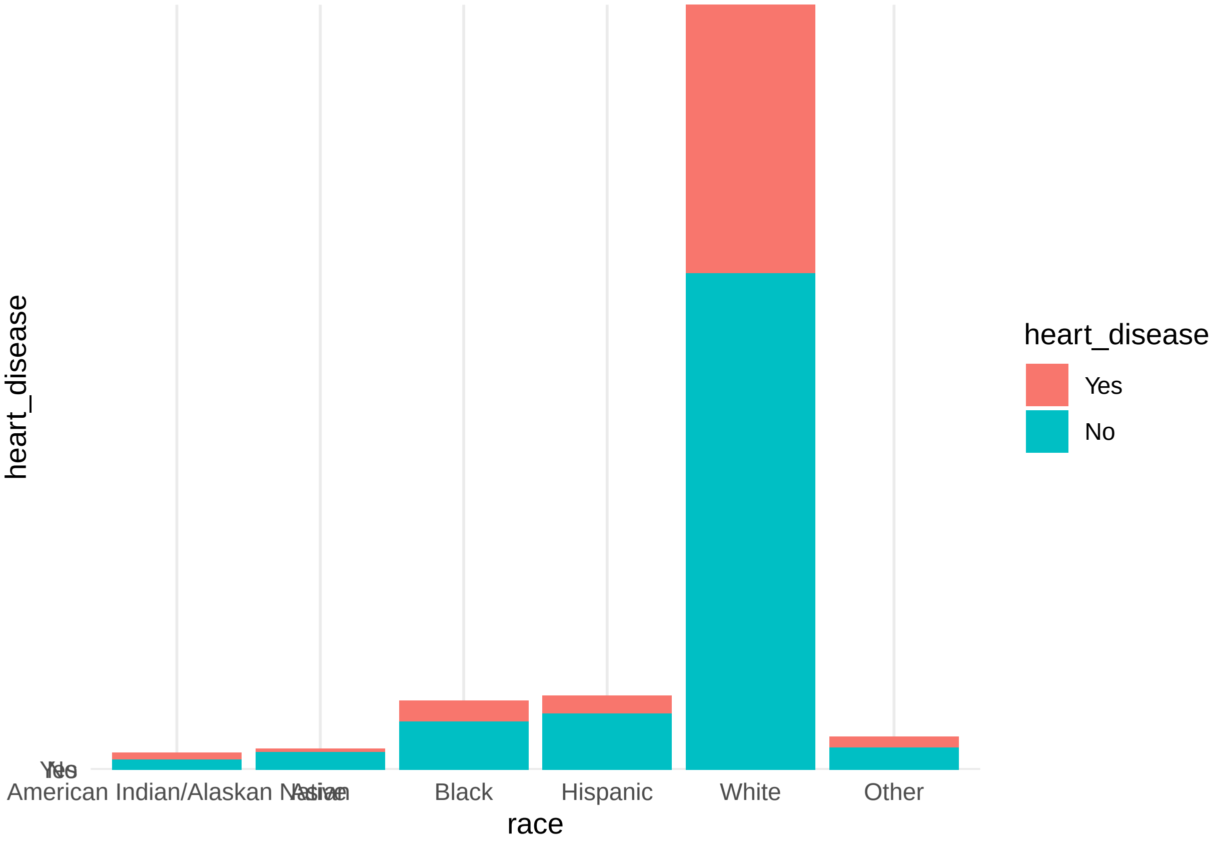
- First, draw a plot of variable `race`.

```
newrecords %>%
  ggplot(aes(x = race)) +
  geom_bar()
```



- According to the graph, we find that most of the respondents are white. For this reason, it maybe hard for us to find the relationship between `race`` and`heart_disease‘.
- Second, draw a plot of variable `race` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= race, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```

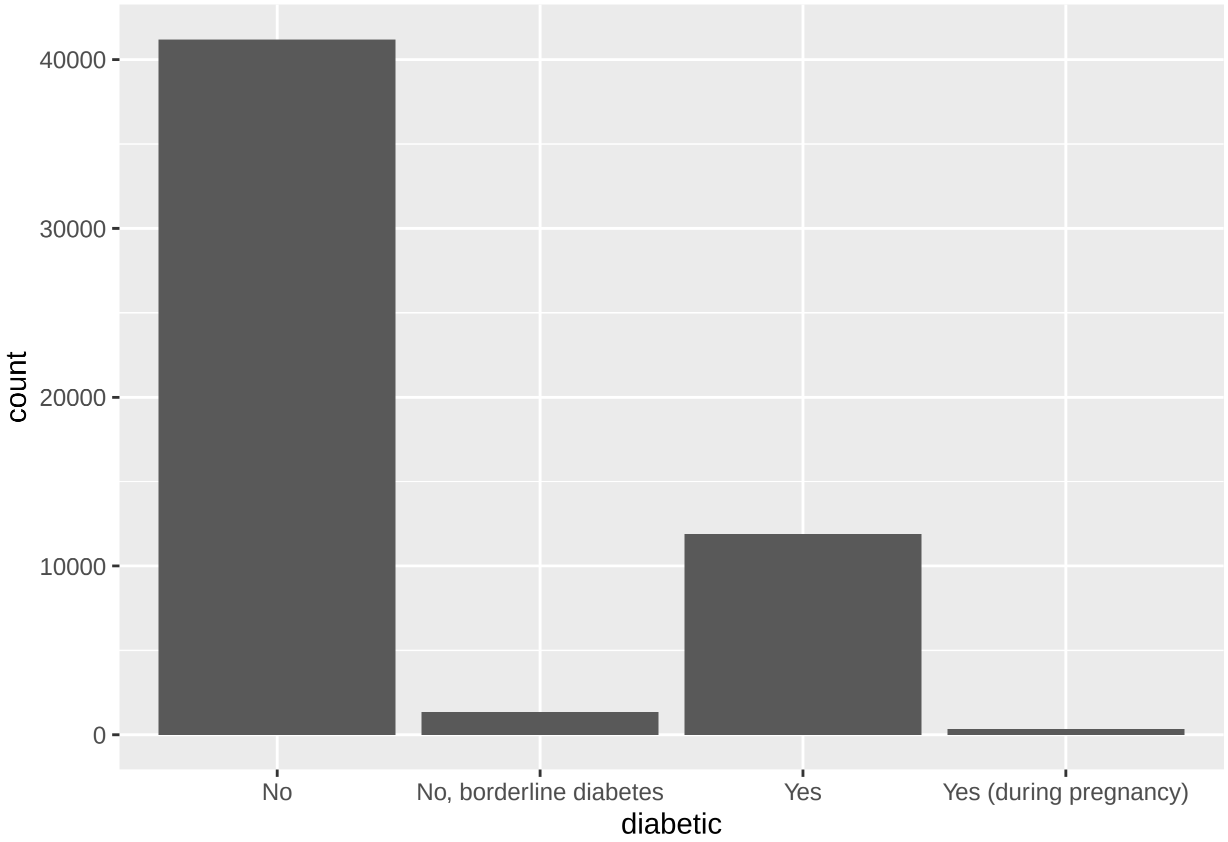


- Based on the graph, although it is difficult for us to tell which race is more likely to heart disease, we can find that the probability of heart disease of different races is not the same. (We will confirm this result at the end of this project)

Variable diabetic

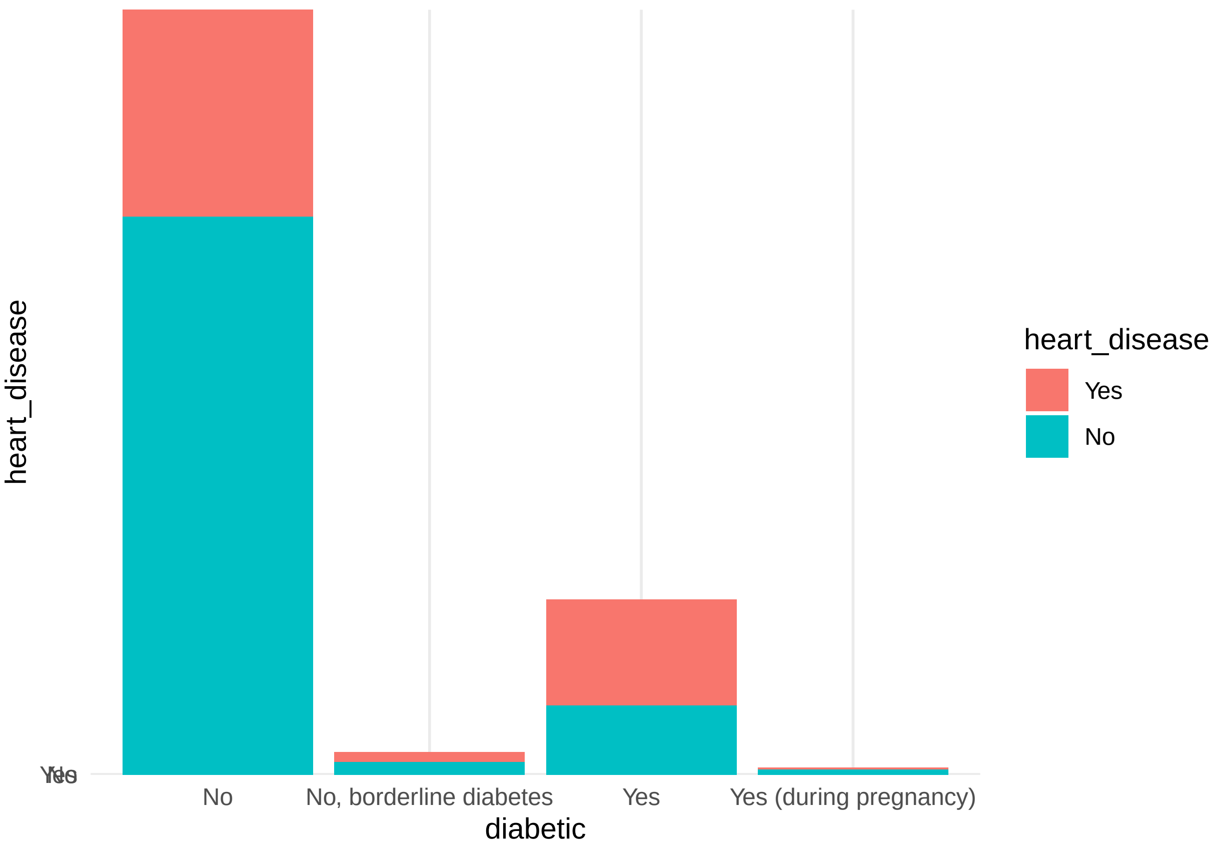
- First, draw a plot of variable `diabetic`.

```
newrecords %>%
  ggplot(aes(x = diabetic)) +
  geom_bar()
```



- According to the graph, we find that most of the respondents are on level ‘No’, and some of the respondents are on level ‘Yes’. For this reason, it maybe hard for us to find the relationship between `diabetic` and `heart_disease`.
- Second, draw a plot of variable `diabetic` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= diabetic, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```



- Based on the graph, we can find that the respondent who had diabetic is more likely to get heart disease. (We will confirm this result at the end of this project)

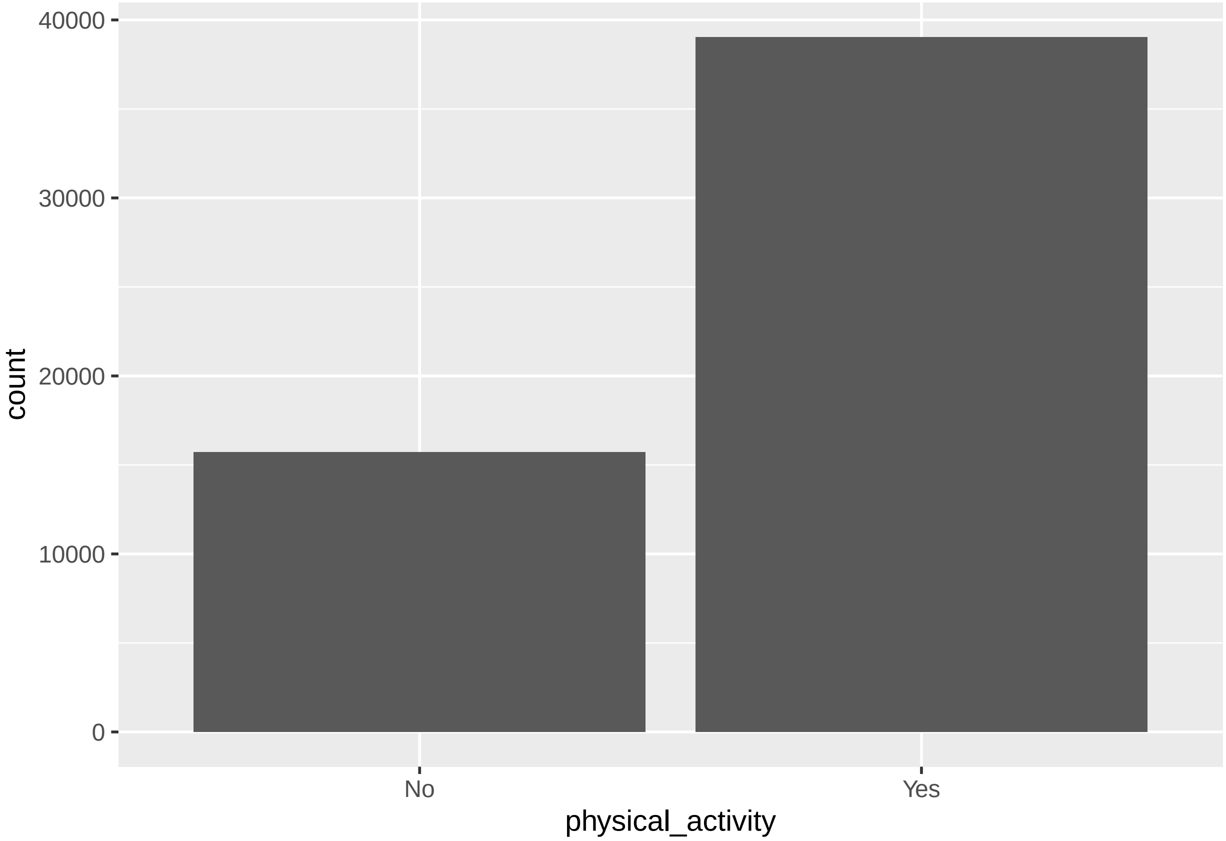
```
head(newrecords)
```

```
##   heart_disease   bmi smoking alcohol_drinking stroke physical_health
## 1            No 33.84      No               No     No               0
## 2            No 31.75      No              Yes     No               0
## 3            No 33.64      No               No     No               0
## 4            No 24.56      No               No     No               0
## 5            No 40.69     Yes               No     No              30
## 6            No 27.89      No               No     No               0
##   mental_health diff_walking    sex age_category  race diabetic
## 1             2           No Female        45-49 White       No
## 2             0           No   Male        55-59 White       No
## 3            28           No   Male        40-44 Black      Yes
## 4             0           No Female        40-44 Asian       No
## 5             0          Yes   Male        60-64 White      Yes
## 6             0           No   Male        75-79 White       No
##   physical_activity gen_health sleep_time asthma kidney_disease skin_cancer
## 1               Yes  Very good          6     No             No         Yes
## 2               Yes  Excellent          7     No             No         Yes
## 3               Yes       Good          7     No             No          No
## 4               Yes  Excellent          6     No             No          No
## 5               Yes       Fair          7     No             No          No
## 6               Yes  Very good          5     No             No         Yes
```

Variable physical_activity

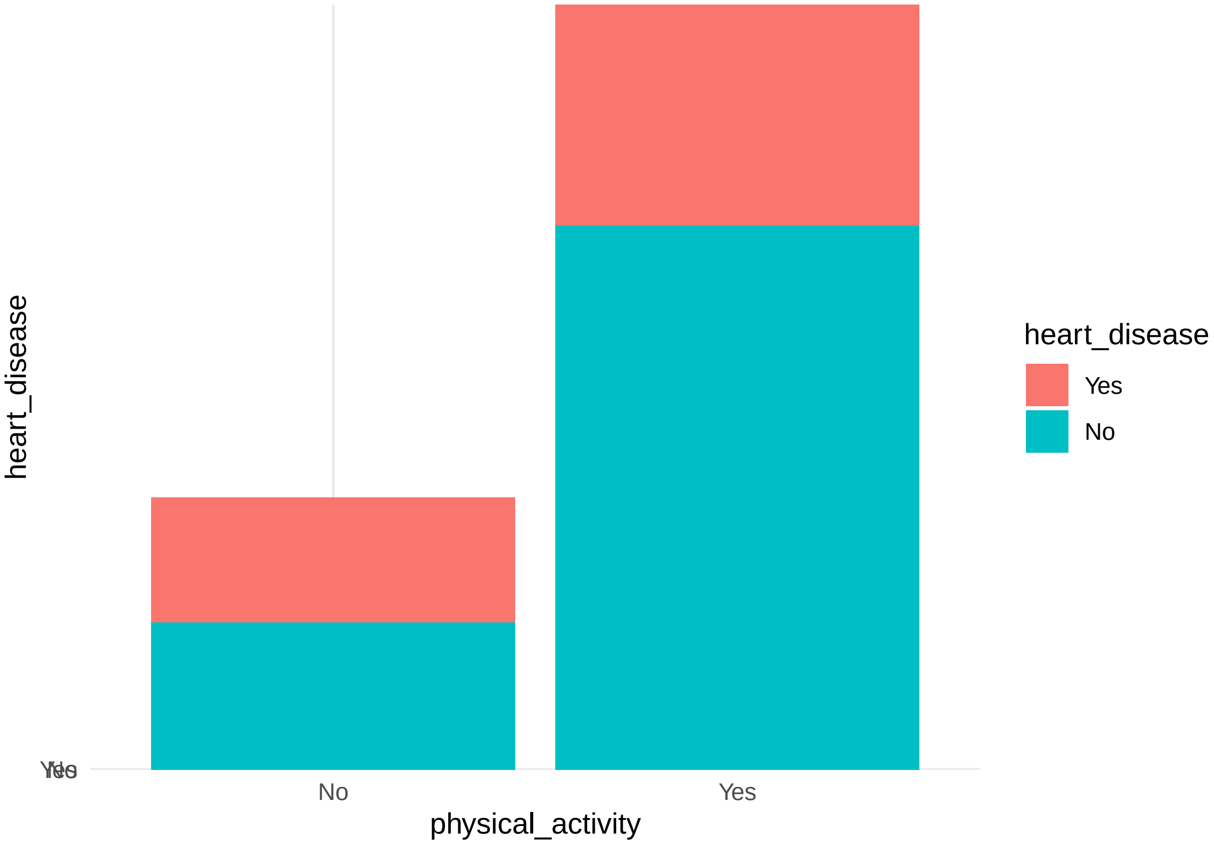
- First, draw a plot of variable physical_activity. physical_activity: Whether the respondent did physical activity or exercise during the past 30 days other than their regular job.

```
newrecords %>%
  ggplot(aes(x = physical_activity)) +
  geom_bar()
```



- According to the graph, we find that most of the respondents are on level 'Yes', and some of the respondents are on level 'No'. For this reason, it maybe hard for us to find the relationship between physical_activity and heart_disease.
- Second, draw a plot of variable physical_activity by heart_disease

```
newrecords %>%
  ggplot(aes(x= physical_activity, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```

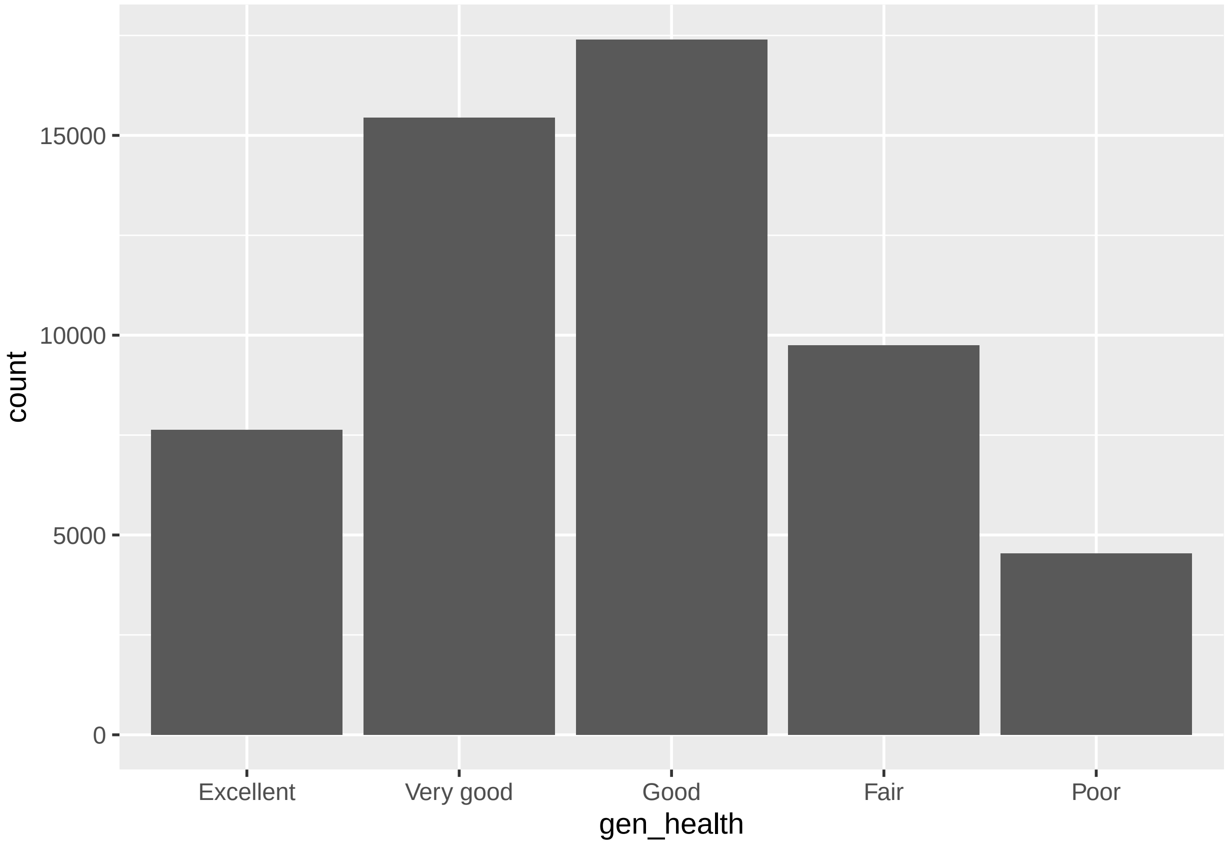


- Based on the graph, we can find that the respondent who didn't do physical activity or exercise during the past 30 days other than their regular job is more likely to get heart disease. (We will confirm this result at the end of this project)

Variable gen_health

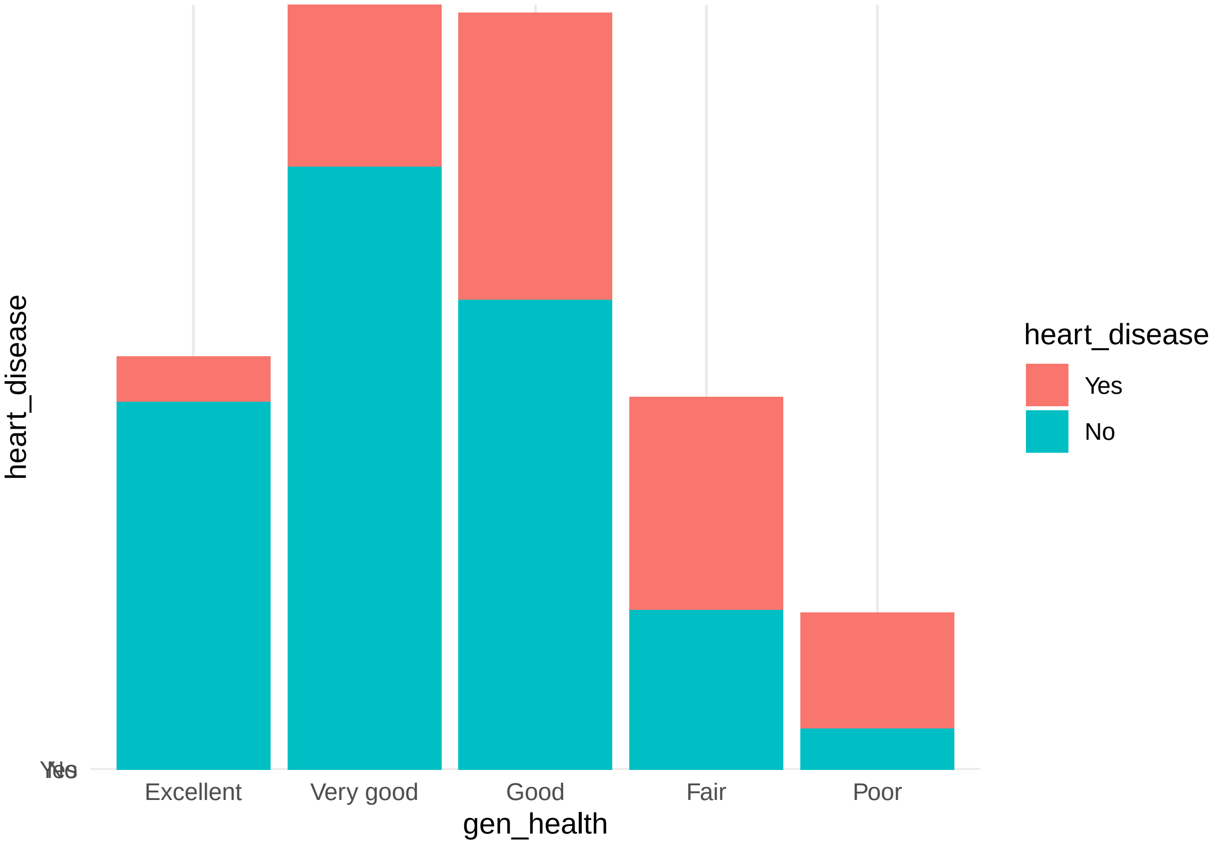
- First, draw a plot of variable `gen_health`. `gen_health`: The respondent's health assessment of his/her self in general [Notes : the answer should be 'Very good', 'Good', 'Excellent', 'Fair', 'Poor']

```
newrecords %>%
  ggplot(aes(x = gen_health)) +
  geom_bar()
```



- According to the plot, we can see that most of respondents think they are in good and above health, and there are some respondents think they are in fair or poor health.
- Second, draw a plot of variable `gen_health` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= gen_health, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```

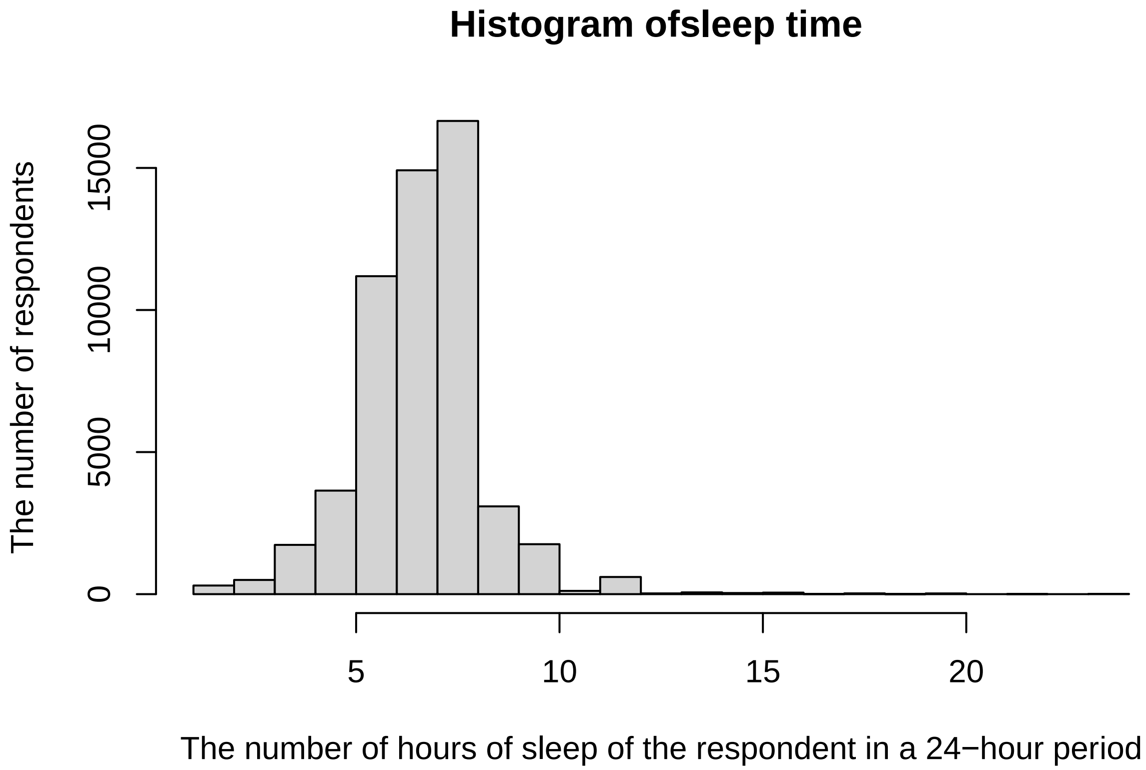


- Based on the graph, we can find that the respondent who think he/she are in poor health is more likely to get heart disease. (We will confirm this result at the end of this project)

Variable sleep_time

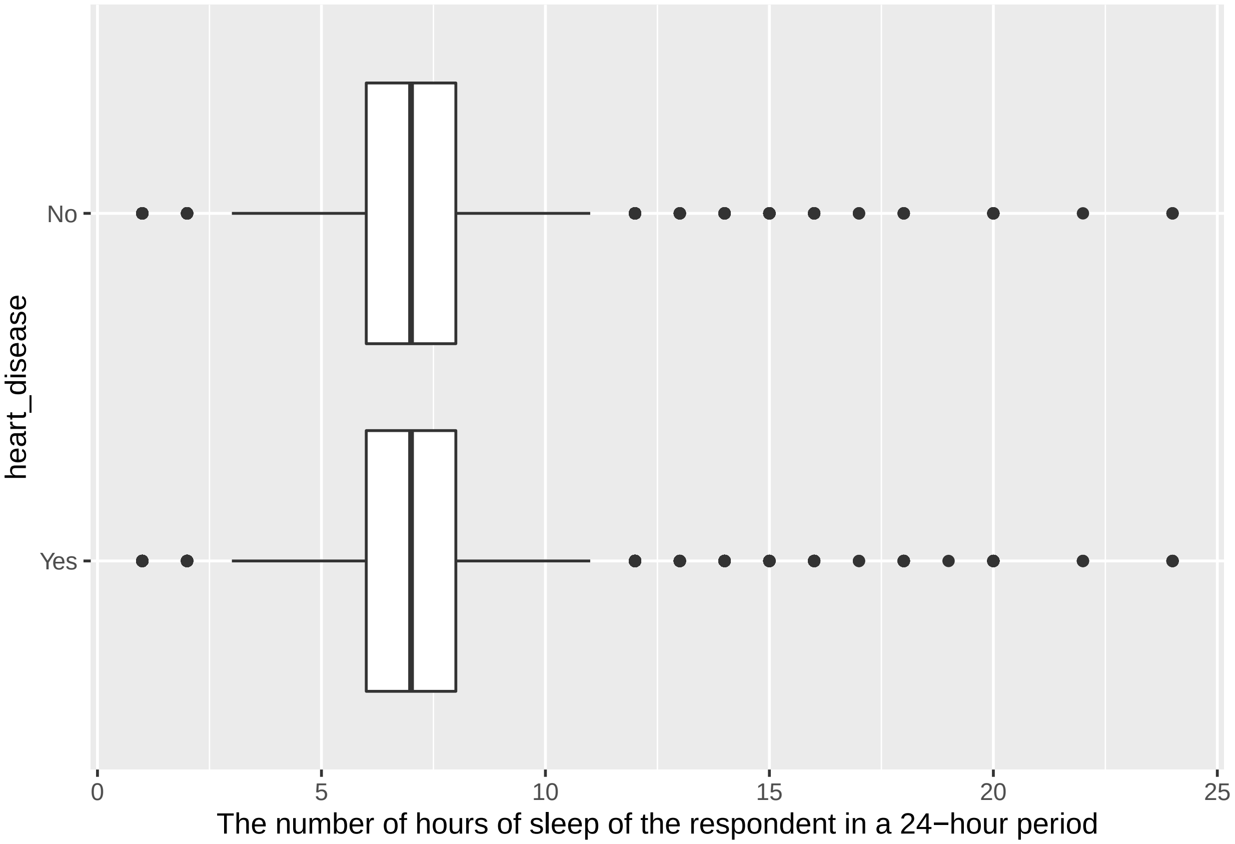
- First, draw a histogram of variable `sleep_time`

```
hist(newrecords$sleep_time, main = paste("Histogram ofsleep time "), xlab = 'The number of hours of slee
```



- According to the graph, we know that the distribution of `sleep_time` definitely appears to be left skewed, and it has a long right tail. It also almost looks a normal distribution. There's one peak around 7-8 hour. Most people have a sleep time between 5- 8 hour.
- Second, draw a boxplot of variable `sleep_time` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x = sleep_time, y=heart_disease))+
  geom_boxplot() +
  xlab("The number of hours of sleep of the respondent in a 24-hour period")
```

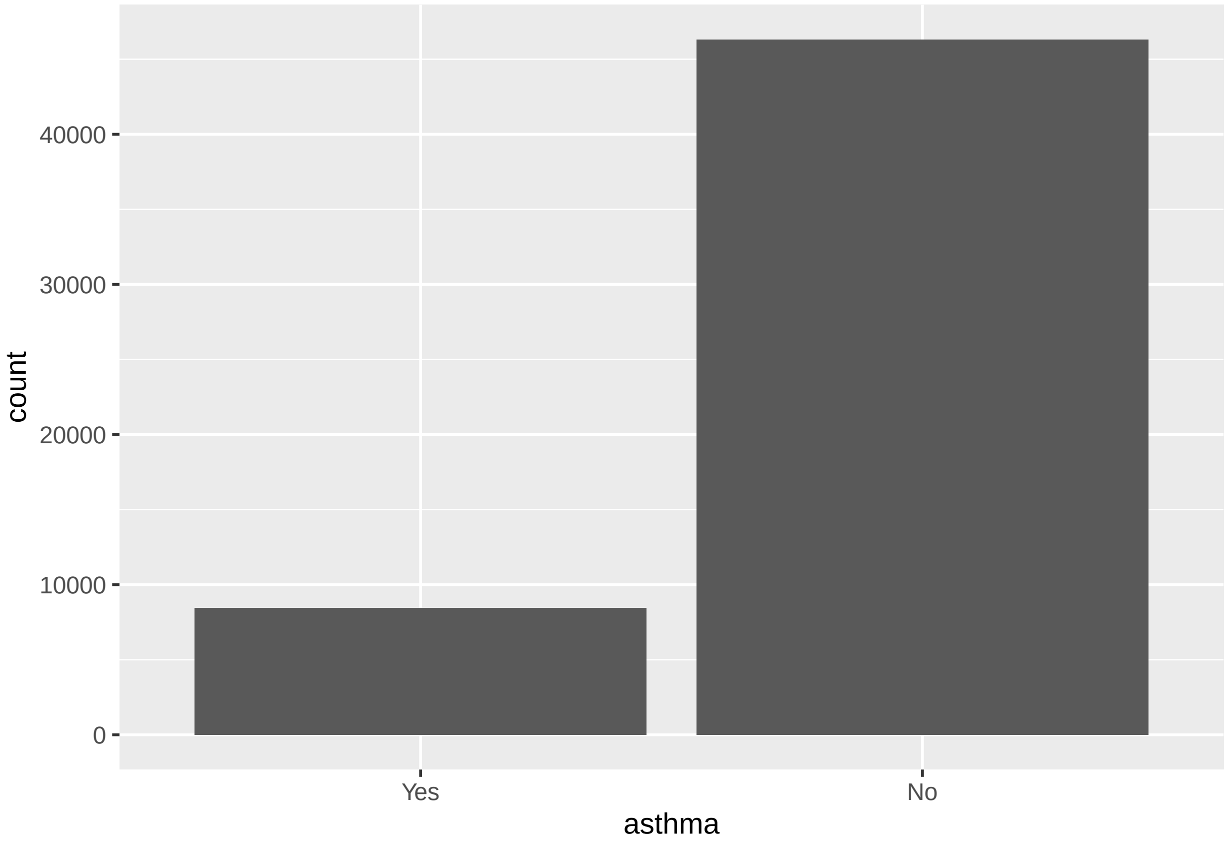


- From the graph we got, since the boxplots of the two levels ('Yes', 'No') are very similar and the medians are very close to each other, it is very hard for us to tell whether the length of the sleep time is related to heart disease or not. We will explore it more at the modeling part.

Variable asthma

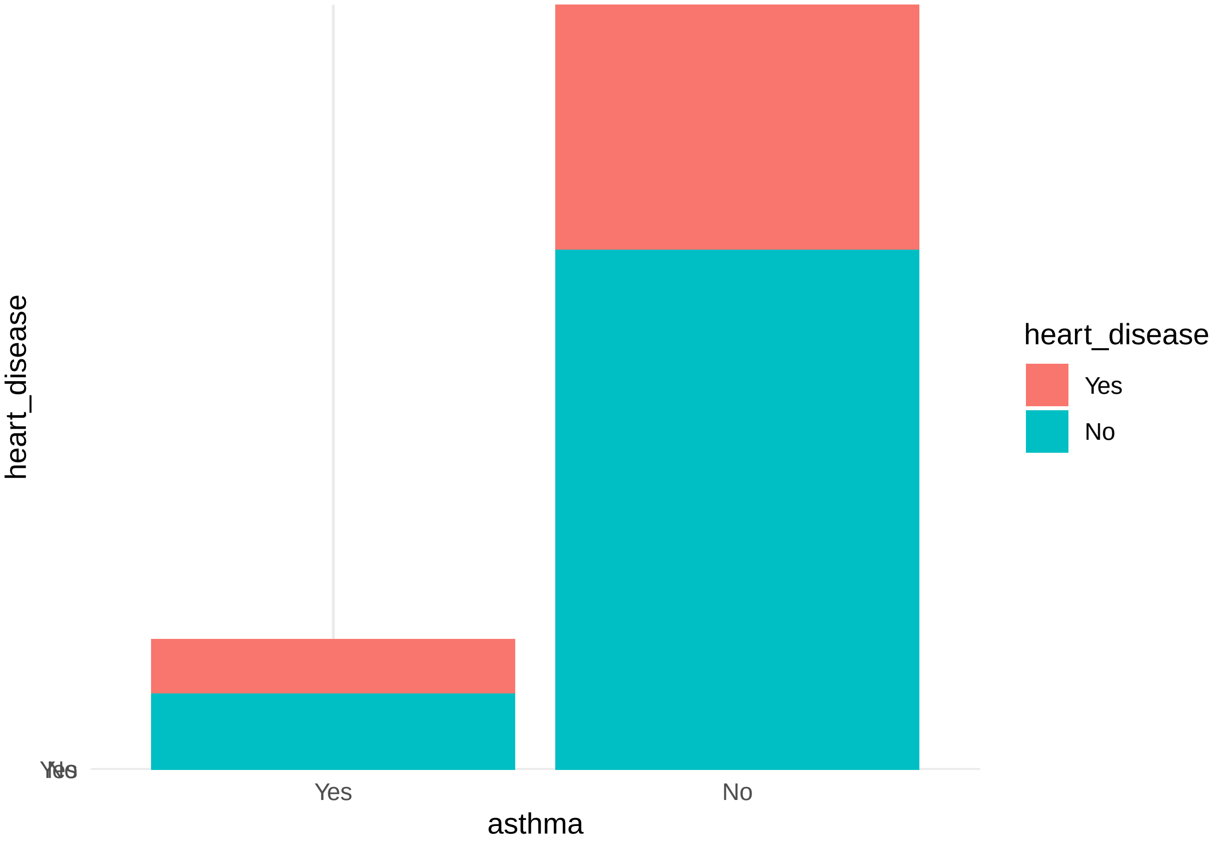
- First, draw a plot of variable asthma

```
newrecords %>%
  ggplot(aes(x = asthma)) +
  geom_bar()
```



- According to the graph, we find that there are much more observations on ‘No’ levels than ‘Yes’ levels for variable `asthma`.For this reason, it maybe hard for us to find the relationship between `asthma` and `heart_disease`.
- Second, draw a plot of variable `asthma` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= asthma, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```



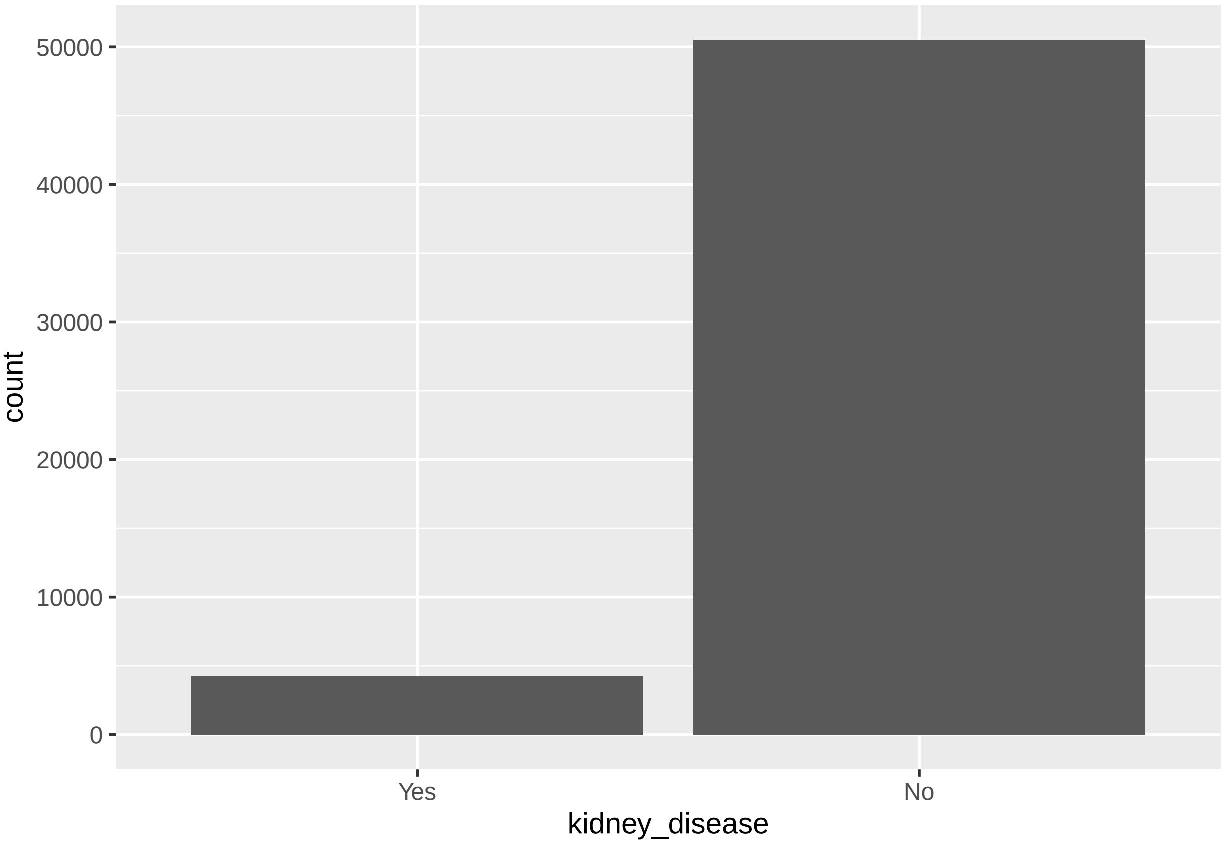
*

Based on the graph, we can find that the respondent who had asthma is more likely to get heart disease. (We will confirm this result at the end of this project)

Variable kidney_disease

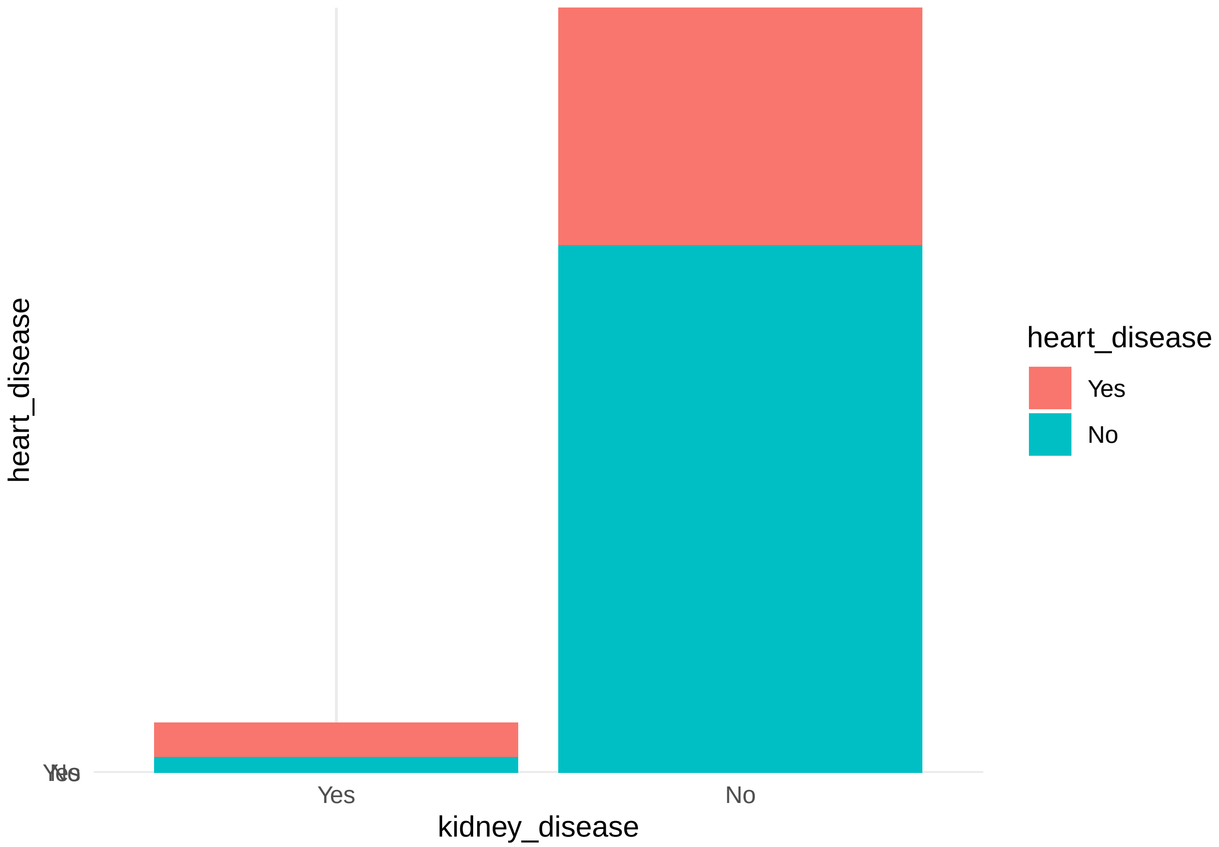
- First, draw a plot of variable kidney_disease

```r
newrecords %>%
  ggplot(aes(x = kidney_disease)) +
  geom_bar()
```



- According to the graph, we find that there are much more observations on ‘No’ levels than ‘Yes’ levels for variable `kidney_disease`.For this reason, it maybe hard for us to find the relationship between `kidney_disease` and `heart_disease`.
- Second, draw a plot of variable `kidney_disease` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= kidney_disease, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```



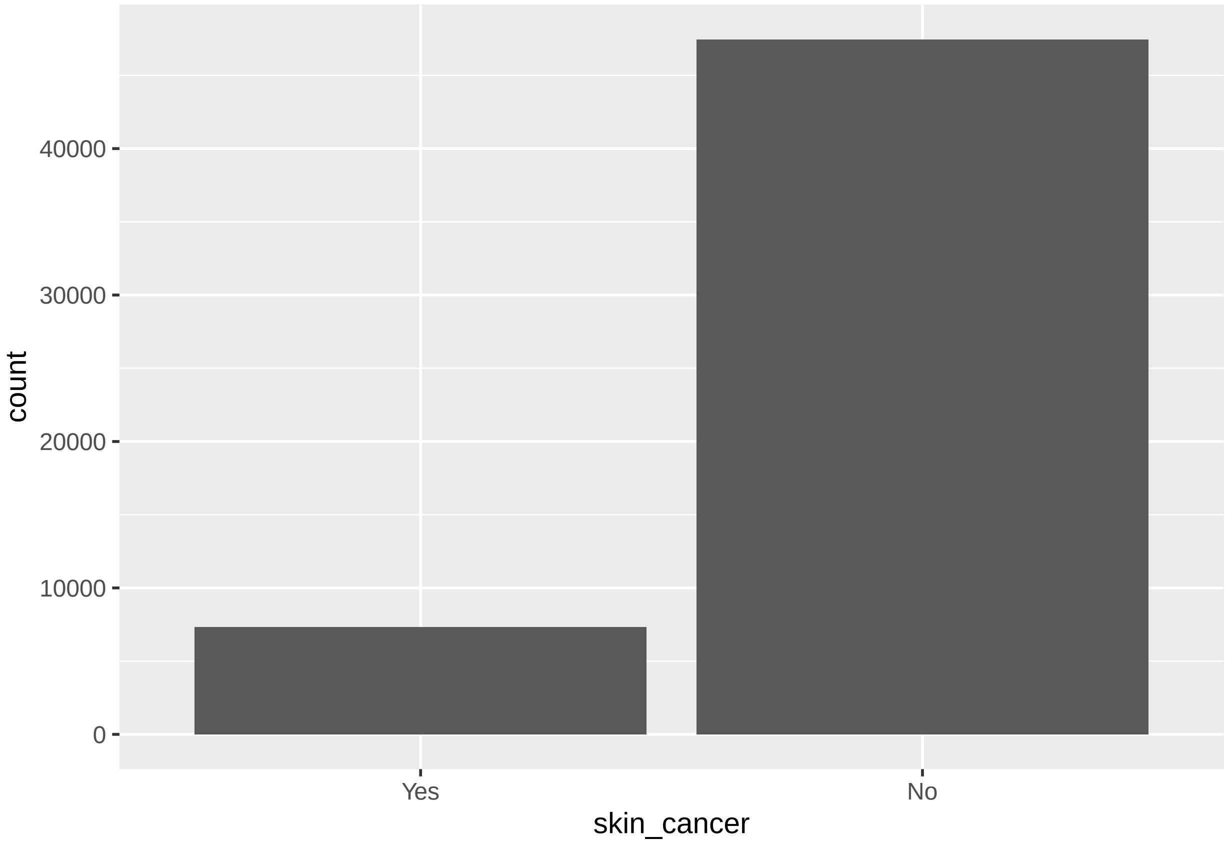
*

Based on the graph, we can find that the respondent who had kidney_disease is more likely to get heart disease. (We will confirm this result at the end of this project)

Variable skin_cancer

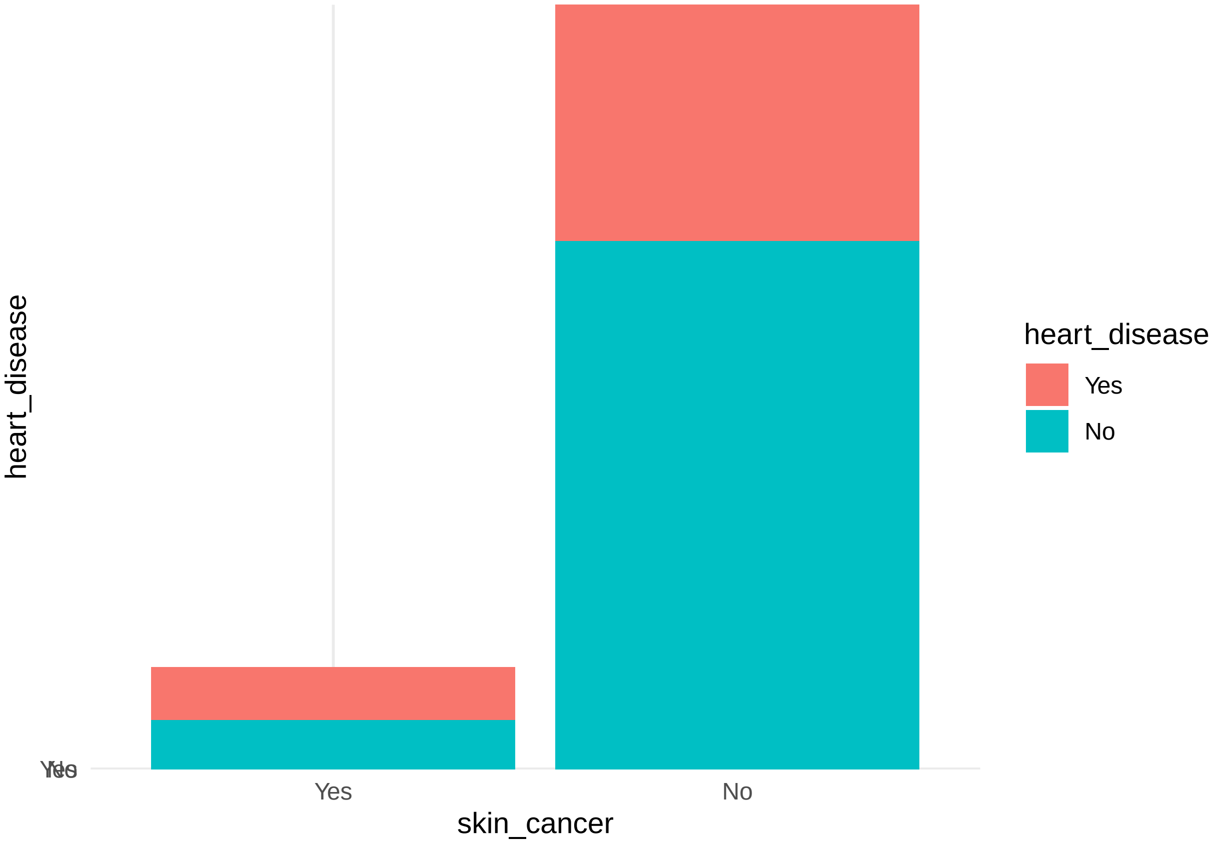
- First, draw a plot of variable skin_cancer

```
newrecords %>%
  ggplot(aes(x = skin_cancer)) +
  geom_bar()
```



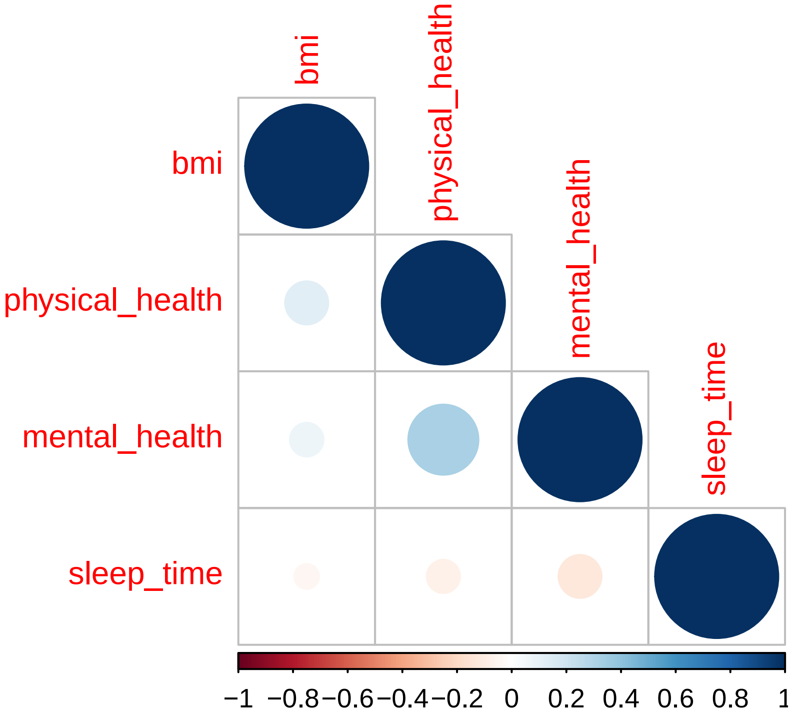
- According to the graph, we find that there are much more observations on 'No' levels than 'Yes' levels for variable `skin_cancer`.For this reason, it maybe hard for us to find the relationship between `skin_cancer` and `heart_disease`.
- Second, draw a plot of variable `skin_cancer` by `heart_disease`

```
newrecords %>%
  ggplot(aes(x= skin_cancer, y= heart_disease , fill= heart_disease)) +
  geom_bar(stat="identity")+theme_minimal()
```



- Based on the graph, we can find that the respondent who had skin_cancer is more likely to get heart disease. (We will confirm this result at the end of this project)

```
newrecords %>%
  select(is.numeric) %>%
  cor() %>%
  corrplot(type = "lower")
```



Data Split

The data was split in a 80% training, 20% testing split. Stratified sampling was used as the `heart_disease` distribution was skewed. (See more on that in the EDA).

The data split was conducted prior to the EDA as I did not want to know anything about my testing data set before I tested my model on those observations.

```
set.seed(1234)
newrecords_split <- newrecords %>%
  initial_split(prop = 0.8, strata = "heart_disease")

newrecords_train <- training(newrecords_split)
newrecords_test <- testing(newrecords_split)
```

```
# check whether the training and testing data sets
# have the appropriate number of observations.
dim(newrecords)
```

```
## [1] 54760    18
```

```
dim(newrecords_train)
```

```
## [1] 43807    18
```

```
dim(newrecords_test)
```

```
## [1] 10953    18
```

- The training data set has about $54760*0.80 = 43808$ observations and the testing data set has just under $54760*0.20 = 10952$ observations. So, according to what we got from the code and our calculations, we can conclude that our training and testing data sets have the appropriate number of observations.

Model Fitting

In this part, I decided to use 4 different model classes (6 models in total).

- Class 1: Logistic regression, LDA and QDA
- Class 2: Boosted tree
- Class 3: Random forest
- Class 4: Nearest Neighbors

Building the Recipe

- Using the training data, create a recipe predicting the outcome variable heart_disease. Include the following predictors:bmi,smoking,alcohol_drinking,stroke,physical_health, mental_health, diff_walking, sex, age_category,race,diabetic, physical_activity,gen_health ,sleep_time,asthma, kidney_disease,skin_cancer
- Dummy-code `smoking`, `alcohol_drinking`,`stroke`,`diff_walking`, `sex`, `age_category`, `race`, `diabetic`, `physical_activity`, `gen_health`, `asthma`,`kidney_disease`, `skin_cancer`; (According to the lecture, we should encode all nominal predictors.)
- Center and scale all predictors.

```
recipe <- recipe(
  heart_disease ~ smoking+ alcohol_drinking+ stroke,physical_health+ mental_health+diff_walking+sex+age_
  step_dummy(all_nominal_predictors()) %>%
  step_center(all_predictors()) %>%
  step_scale(all_predictors())
```

Uses cross-validation to fold training set.

- We use v-fold cross-validation on the training set. Use 5 folds.
- We stratify the folds by heart_disease as well.
- Stratifying the folds can be useful since it can makes sure the distribution of a heart_disease (often the outcome) remains the same across resamples or, in cross-validation, across folds.

```
newrecords_folds <- vfold_cv(data = newrecords_train, v = 5, strata = heart_disease )
```

Class 1: Logistic regression, LDA and QDA

Since these three models belong to the same model class, we want to select the best model among the three models to represent the best model in this class. We will finally use the four models from 4 different classes to select the best model of the four model classes as our final model.

Logistic regression We specify a logistic regression model for classification using the "glm" engine. Then create a workflow. After that, we add model and the appropriate recipe (we created before).

```
log_reg <- logistic_reg() %>%
  set_engine("glm") %>%
  set_mode("classification")

log_wkflow <- workflow() %>%
  add_model(log_reg) %>%
  add_recipe(recipe)
```

LDA In a similar process, but this time specify a linear discriminant analysis model for classification using the "MASS" engine.

```
lda_mod <- discrim_linear() %>%
  set_mode("classification") %>%
  set_engine("MASS")

lda_wkflow <- workflow() %>%
  add_model(lda_mod) %>%
  add_recipe(recipe)
```

QDA In a similar process, but this time specify a quadratic discriminant analysis model for classification using the "MASS" engine.

```
qda_mod <- discrim_quad() %>%
  set_mode("classification") %>%
  set_engine("MASS")

qda_wkflow <- workflow() %>%
  add_model(qda_mod) %>%
  add_recipe(recipe)
```

Assess the performance of each of these three models

- Fit each of the models created before to the folded data.

```
control <- control_resamples(save_pred = TRUE)

log_fit <- fit_resamples(log_wkflow, newrecords_folds)

lda_fit <- fit_resamples(resamples = newrecords_folds,
                         lda_wkflow,
                         control = control)

qda_fit <- fit_resamples(qda_wkflow, resamples = newrecords_folds,
                         control = control)
```

- We will use collect_metrics() to print the mean and standard errors of the performance metric accuracy across all folds for each of the four models.
- We will decide which of the 3 fitted models has performed the best.

```
collect_metrics(log_fit)
```

```
## # A tibble: 2 x 6
##   .metric  .estimator  mean     n std_err .config
##   <chr>    <chr>      <dbl> <int>   <dbl> <chr>
## 1 accuracy binary     0.627     5 0.00261 Preprocessor1_Model1
## 2 roc_auc  binary     0.650     5 0.00285 Preprocessor1_Model1
```

```
collect_metrics(lda_fit)
```

```
## # A tibble: 2 x 6
##   .metric  .estimator  mean     n std_err .config
##   <chr>    <chr>      <dbl> <int>   <dbl> <chr>
## 1 accuracy binary     0.627     5 0.00261 Preprocessor1_Model1
## 2 roc_auc  binary     0.651     5 0.00301 Preprocessor1_Model1
```

```
collect_metrics(qda_fit)
```

```
## # A tibble: 2 x 6
##   .metric  .estimator  mean     n  std_err .config
##   <chr>    <chr>      <dbl> <int>    <dbl> <chr>
## 1 accuracy binary     0.567     5 0.000244 Preprocessor1_Model1
## 2 roc_auc  binary     0.650     5 0.00213  Preprocessor1_Model1
```

Based on the results we get here, we can see that these three models have very similar standard error of the accuracy (and the differences between them are very small). Since the mean accuracy of the Logistic regression equals to the mean accuracy of the LDA and both are higher than the mean accuracy of QDA, we know that logistic regression and LDA are better than QDA. However, since the LDA model also has the highest mean roc_auc value in these three models. We can conclude that LDA is the best model in these three models.

- Now that we've chosen a model, fit our chosen model to the entire training dataset (not to the folds).

```
lda_fit_train <- fit(lda_wkflow, newrecords_train)
```

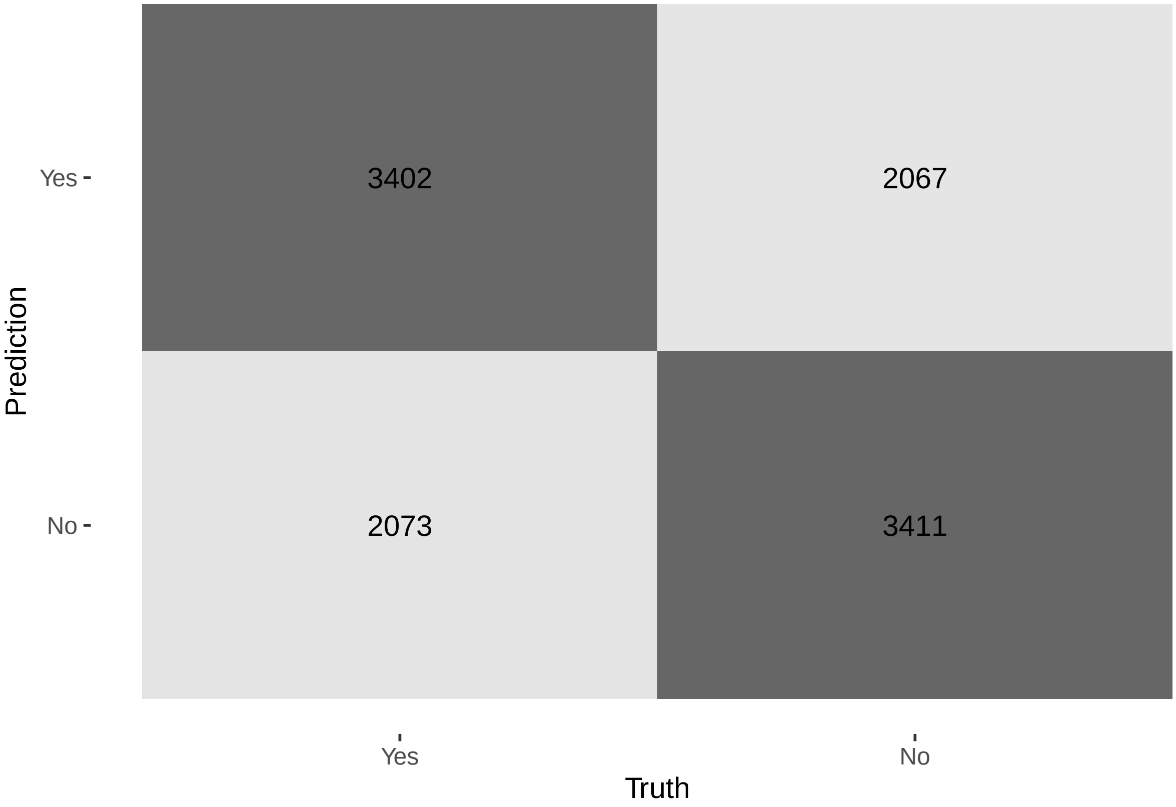
- Finally, with your fitted model, use predict(), bind_cols(), and accuracy() to assess your model's performance on the testing data!

```
lda_test <- fit(lda_wkflow, newrecords_test)
predict(lda_test, new_data = newrecords_test, type = "class") %>%
  bind_cols(newrecords_test %>% select(heart_disease)) %>%
  accuracy(truth = heart_disease, estimate = .pred_class)
```

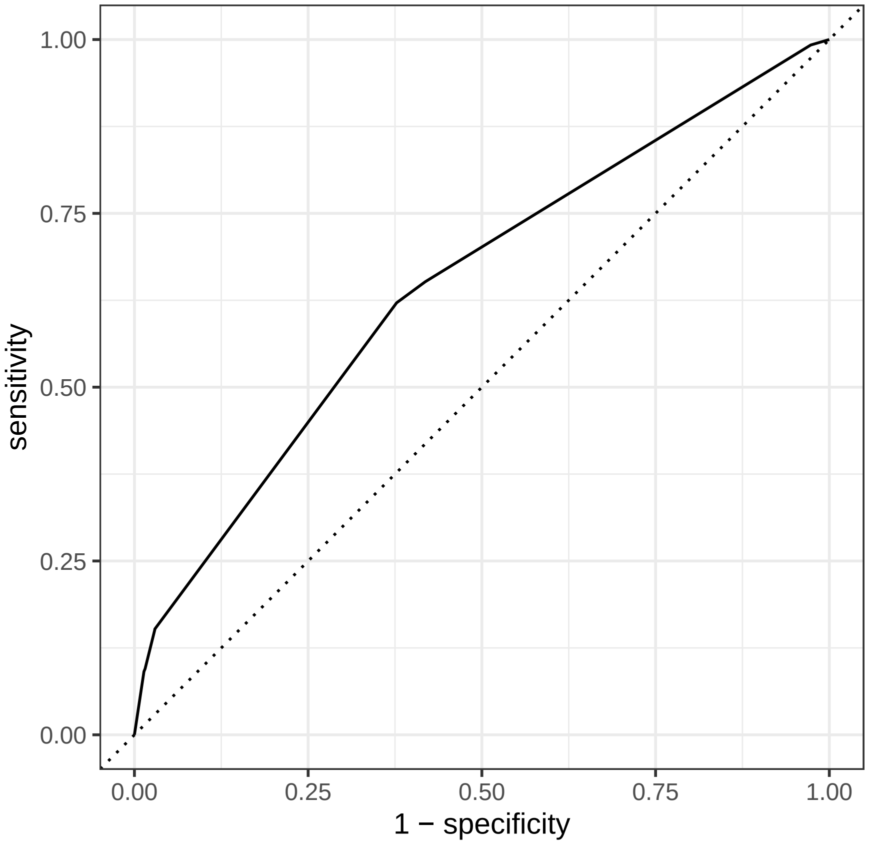
```
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy binary         0.622
```

Confusion matrix, ROC curve and AUC values Now, using the testing data, we want to create a confusion matrix and visualize it. Plot an ROC curve and calculate the area under it (AUC).

```
#  create a confusion matrix and visualize it
augment(lda_test, new_data = newrecords_test) %>%
  conf_mat(truth = heart_disease, estimate = .pred_class) %>%
  autoplot(type = "heatmap")
```



```
# Plot roc_curve
augment(lda_test, new_data = newrecords_test) %>%
  roc_curve(heart_disease, .pred_Yes) %>%
  autoplot()
```



```r
# Calculate AUC
augment(lda_test, new_data = newrecords_test) %>%
  roc_auc(heart_disease, .pred_Yes)
```

```
## # A tibble: 1 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 roc_auc binary         0.646
```

Based on the results, we find that although LDA is the best among the 3 models, it is still not good enough. We want to see if there is a model work better than LDA in other model class.

Class 2 : Boosted tree