Heart Disease Report

Humberto Garza

21/NOV/2022

Contents

1 Introduction

Heart disease is regarded as "the leading cause of death in the United States"[1]. Is among the most prevalent chronic diseases in the United States, impacting millions of Americans each year and exerting a significant financial burden on the economy. In the United States alone, heart disease claims roughly 647,000 lives each year[2].

There are many risk factors for heart diseases: age, sex, tobacco use, physical inactivity, non-alcoholic fatty liver disease, excessive alcohol consumption, unhealthy diet, obesity, genetic predisposition and family history of cardiovascular disease, raised blood pressure (hypertension), raised blood sugar (diabetes mellitus), raised blood cholesterol (hyperlipidemia), undiagnosed celiac disease, psychosocial factors, poverty and low educational status, air pollution, and poor sleep.[3]

The symptoms of heart disease will depend on the type of heart disease you have. You may not have symptoms at first. In some cases, you may not know you have heart disease until you have a complication such as a heart attack.[4] This fact highlights the importance of preventative measures and tests that can accurately predict heart disease in the population prior to negative outcomes like myocardial infarctions taking place.[2]

Existing cardiovascular disease or a previous cardiovascular event, such as a heart attack or stroke, is the strongest predictor of a future cardiovascular event. Age, sex, smoking, blood pressure, blood lipids and diabetes are important predictors of future cardiovascular disease in people who are not known to have cardiovascular disease. These measures, and sometimes others, may be combined into composite risk scores to estimate an individual's future risk of cardiovascular disease.[5]

The data set used in this project was downloaded from www.kaggle.com, by user "Alex Teboul" who cleaned and synthesized a health-related telephone survey called: "The Behavioral Risk Factor Surveillance System" (BRFSS), that is collected annually by the CDC

In this report we will try to predict heart disease with the data set attributes

This project is for Data Science: Capstone (PH125.9x) course in the HarvardX Professional Certificate Data Science Program

1.1 Data Acquisition

```
if(!require(dslabs)) install.packages("dslabs", repos = "http://cran.us.r-project.org")
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
if(!require(data.table)) install.packages("data.table", repos = "http://cran.us.r-project.org")
if(!require(gridExtra)) install.packages("gridExtra", repos = "http://cran.us.r-project.org")
if(!require(matrixStats)) install.packages("matrixStats", repos = "http://cran.us.r-project.org")
if(!require(xgboost)) install.packages("xgboost", repos = "http://cran.us.r-project.org")

library(dslabs)
library(tidyverse)
library(caret)
library(data.table)
library(gridExtra)
library(matrixStats)
library(xgboost)
```

```
# The heart disease data set is available for download at the following link in csv format:
# https://drive.google.com/file/d/1-UJK8-oj9jrJh1eT5GQfUc1tYKPO94fl/view?usp=sharing

# Of from Github project webpage:
# https://github.com/EDXGarza/HeartDisease_PH125.9x/blob/main/heart_disease.csv

# Or from the original creator of the dataset in zip format:
# https://www.kaggle.com/datasets/alexteboul/heart-disease-health-indicators-dataset/download?datasetVe

# Once downloaded and extracted the data set can be loaded into R with:

hd <- read.csv("/FILE_LOCATION/heart_disease.csv")
```

1.2 Explaining Variables

The selected features for the dataset are:

HeartDiseaseorAttack: Respondents that have ever reported having coronary heart disease or myocardial infarction

HighBP: High Blood Pressure. Adults who have been told they have high blood pressure by a doctor, nurse, or other health professional

HighChol: High Cholesterol. Have you EVER been told by a doctor, nurse or other health professional that your blood cholesterol is high?

CholCheck: Cholesterol check within past five years

BMI: Body Mass Index (BMI)

Smoker: Have you smoked at least 100 cigarettes in your entire life?

Stroke: (Ever told) you had a stroke

Diabetes: (Ever told) you have diabetes

PhysActivity: Physical Activity. Adults who reported doing physical activity or exercise during the past 30 days other than their regular job

Fruits: Consume Fruit 1 or more times per day

Veggies: Consume Vegetables 1 or more times per day

HvyAlcoholConsump: Alcohol Consumption. Heavy drinkers, adult men having more than 14 drinks per week and adult women having more than 7 drinks per week

AnyHealthcare: Do you have any kind of health care coverage, including health insurance, prepaid plans such as HMOs, or government plans such as Medicare, or Indian Health Service?

NoDocbcCost: Was there a time in the past 12 months when you needed to see a doctor but could not because of cost

GenHlth: Health General. Would you say that in general your health is:

MentHlth: Now thinking about your mental health, which includes stress, depression, and problems with emotions, for how many days during the past 30 days was your mental health not good?

PhysHlth: Now thinking about your physical health, which includes physical illness and injury, for how many days during the past 30 days was your physical health not good?

DiffWalk: Do you have serious difficulty walking or climbing stairs?

Sex: Indicate sex of respondent.

Age: Fourteen-level age category

Education: Highest grade or year of school completed

Income: Annual household income from all sources

2 Analysis

2.1 Exploratory Data Analysis

We will start to explore the relation that exist, between each of the variables of the data set and the presence of disease.

Check if there are any missing values in our data set.

```
check_na <- hd %>%  summarize(across(.cols = everything(),
.fns = ~ any(is.na(.x)), .names = "{col}"))
```

HeartDiseaseorAttack	HighBP	HighChol	CholCheck	BMI	Smoker	Stroke	Diabetes
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE

PhysActivity	Fruits	Veggies	HvyAlcoholConsump	AnyHealthcare	NoDocbcCost
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE

GenHlth	MentHlth	PhysHlth	DiffWalk	Sex	Age	Education
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE

The first thing we notice is a huge imbalance in the amount of healthy cases vs disease.

```
tab <- table(hd$HeartDiseaseorAttack)
hd %>% summarize(Disease_cases = sum(HeartDiseaseorAttack == 1),
Healthy_cases = sum(HeartDiseaseorAttack == 0)) %>% knitr::kable()
```

Disease_cases	Healthy_cases
23893	229787

There are 9.62 times more healthy cases than disease cases,

We will balance the data set using random samples and plot the different variables against the disease cases, to examine the relation between them.

```
# Create index of the disease cases
disease_index <- which(hd$HeartDiseaseorAttack == 1)
# Create index of the healthy cases
health_index <- which(hd$HeartDiseaseorAttack == 0)
# Extract a sample of healthy cases the same length as the disease cases
set.seed(9, sample.kind = "Rounding")
ind <- sample(health_index, length(disease_index), replace = FALSE)
# Join the two index variables
balanced_ind <- union(disease_index, ind)
# Make dataset with balanced outcomes
balanced <- hd[balanced_ind,]
# Rename the first column to a shorter name
balanced <- balanced %>% rename(Disease = HeartDiseaseorAttack)
```

The following variables in our data set consist of binary variables:

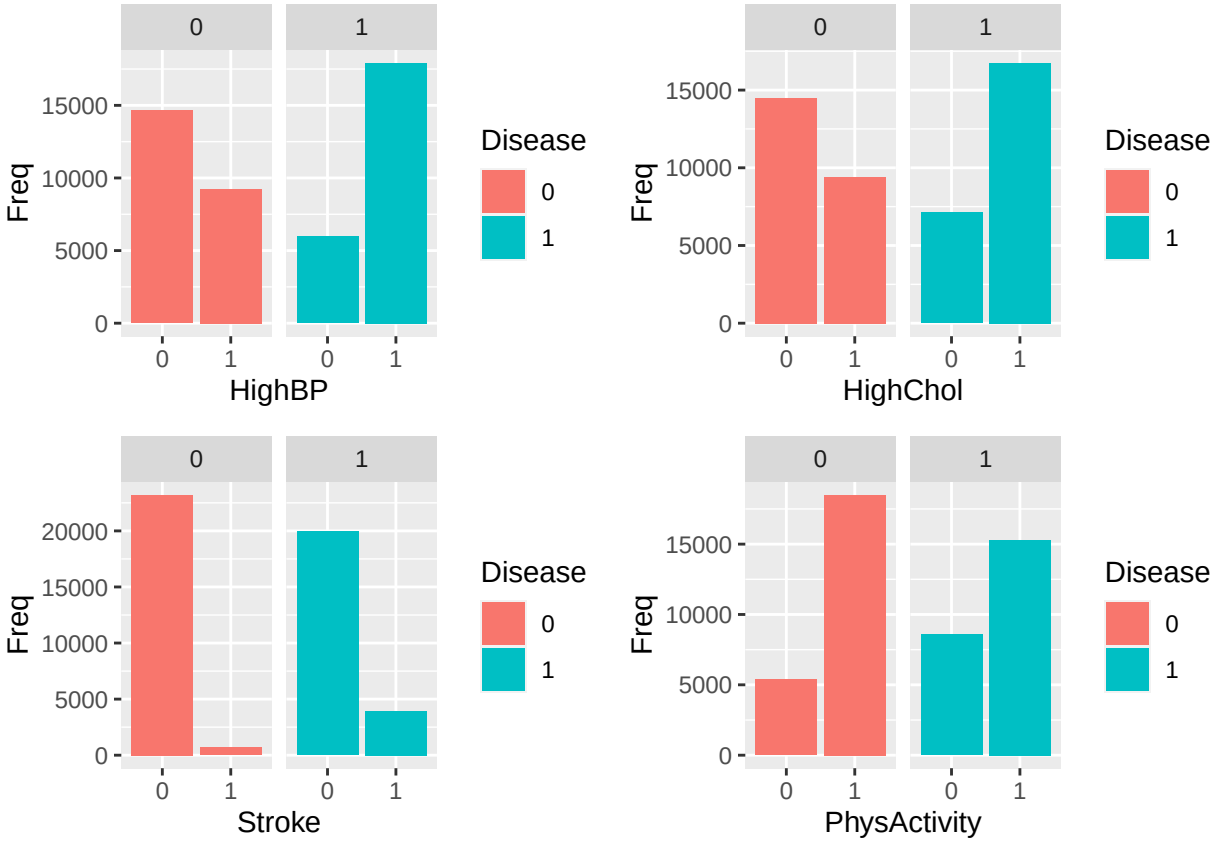
“Disease”, “HighBP”, “HighChol”, “CholCheck”, “Smoker”, “Stroke”, “PhysActivity”, “Fruits”, “Veggies”, “HvyAlcoholConsump”, “AnyHealthcare”, “NoDocbcCost”, “DiffWalk”, “Sex”.

While the rest are numerical discrete:

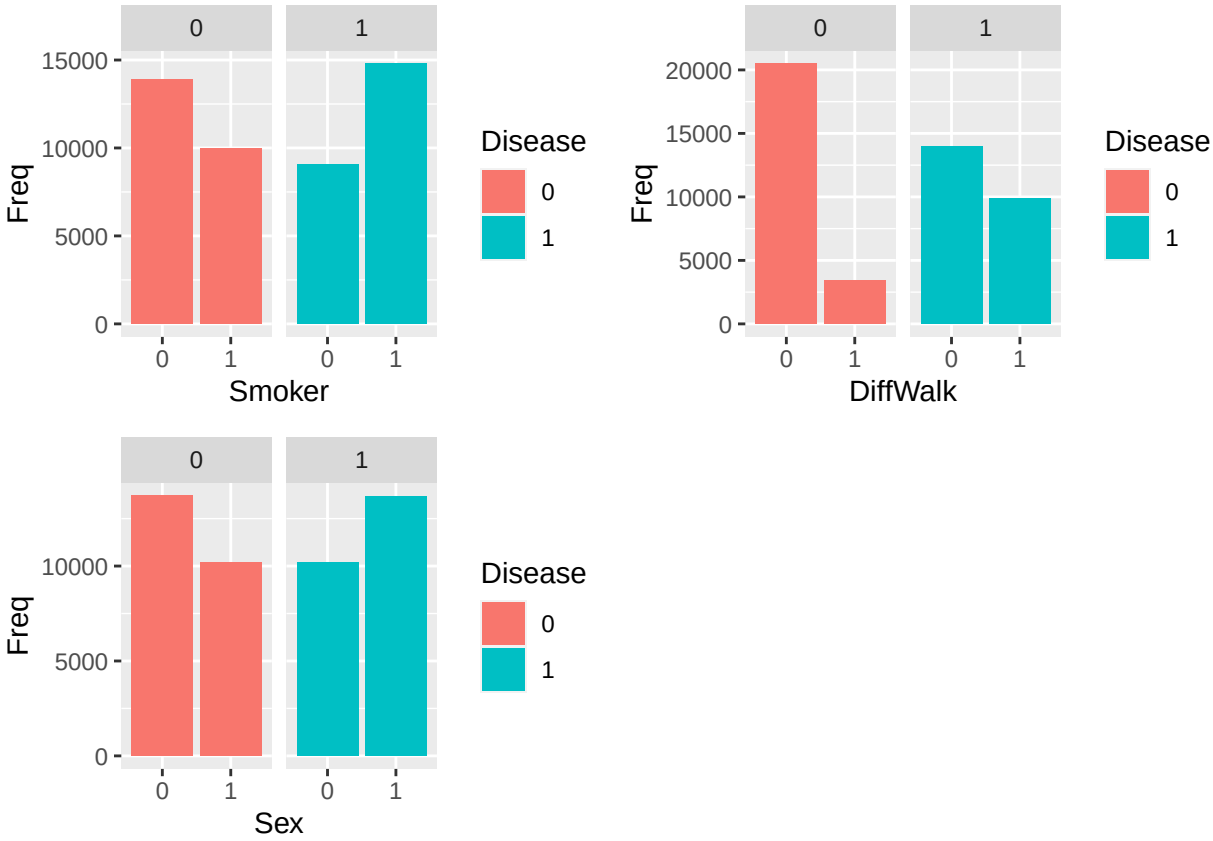
“BMI”, “Diabetes”, “GenHlth”, “MentHlth”, “PhysHlth”, “Age”, “Education”, “Income”.

2.2 Charts assesment

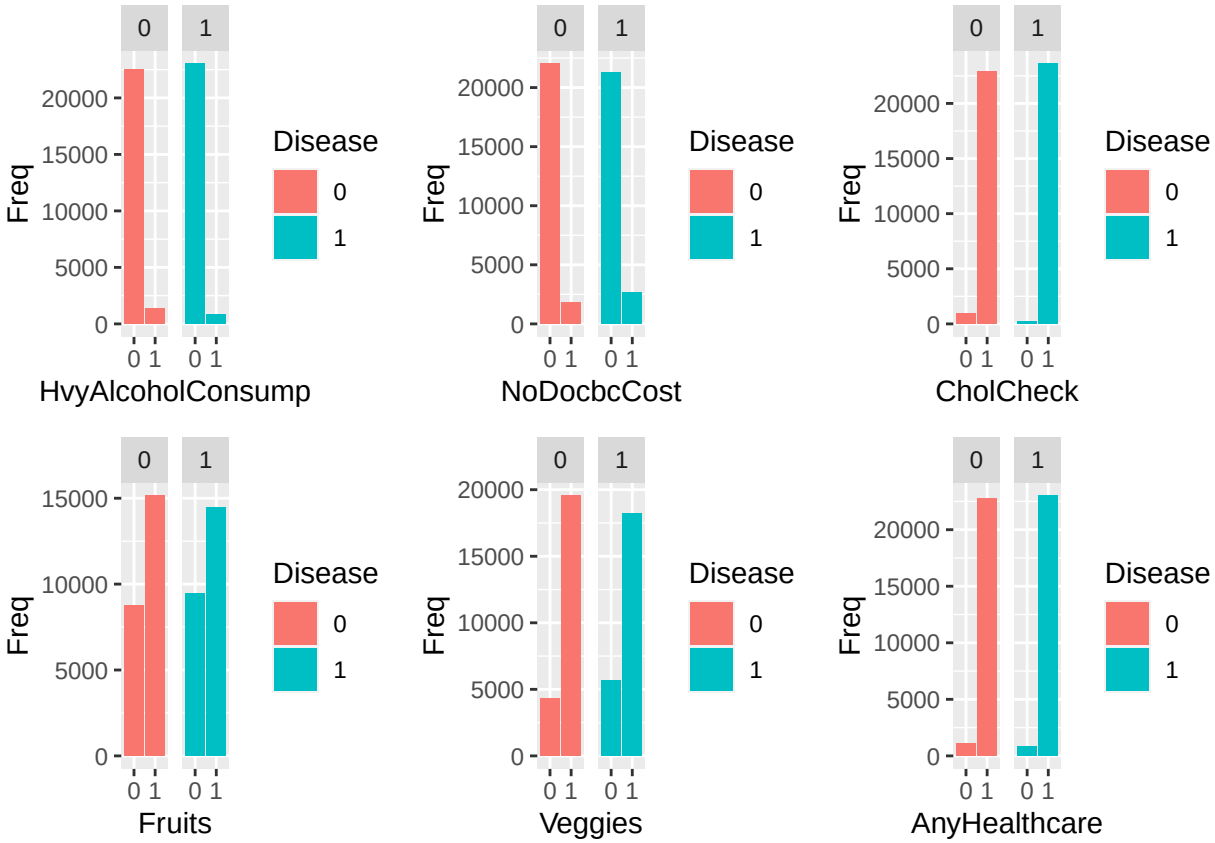
Make bar plots for binary variables:



We get some interesting insights from the bar plots, for an instance, 75.03% of the disease cases present high blood Pressure, similar situation with 70.12% of the disease cases showing high cholesterol

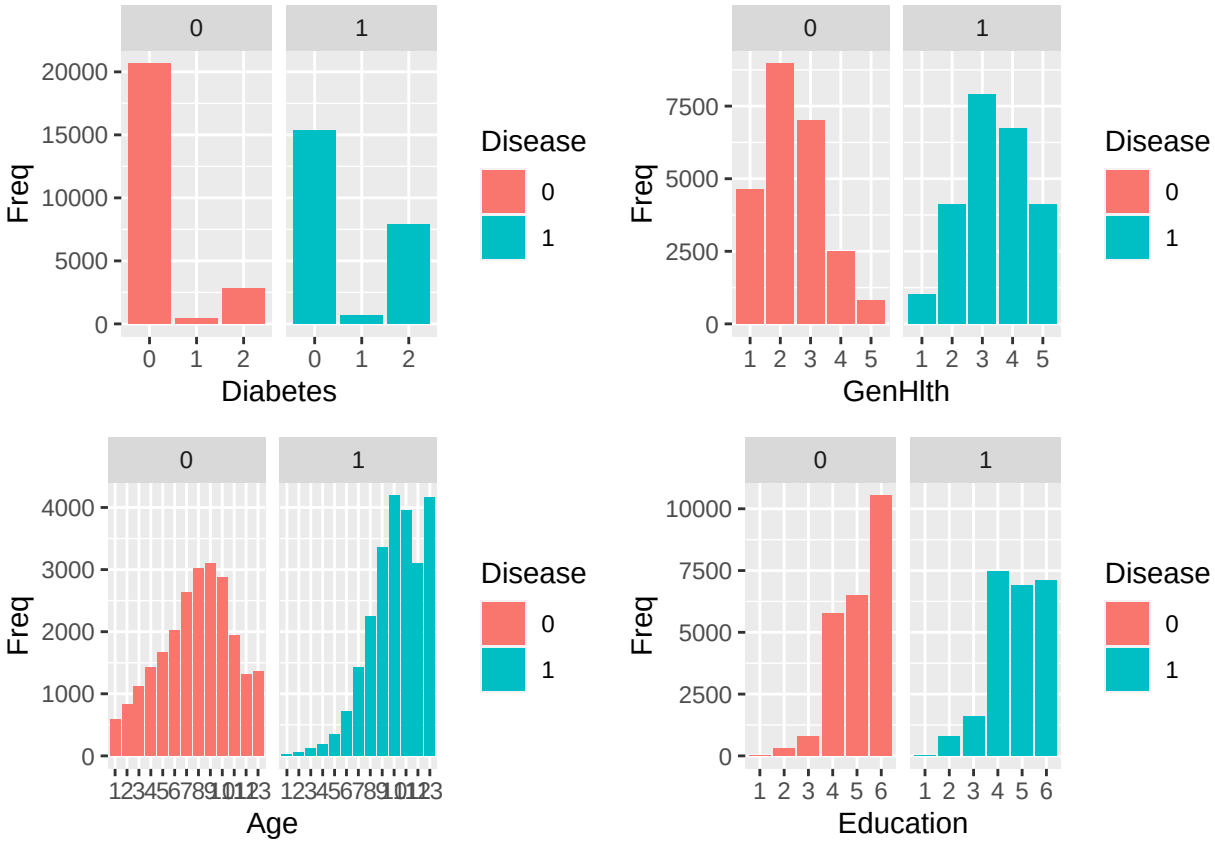


Here we observe the proportion of disease cases that reported being smokers is 61.95%, also heart disease is more prevalent in one gender than the other 57.29%



In general we observe a very similar distribution of cases for the majority of the variables with minor variations.

Next we will examine the numerical discrete variables.

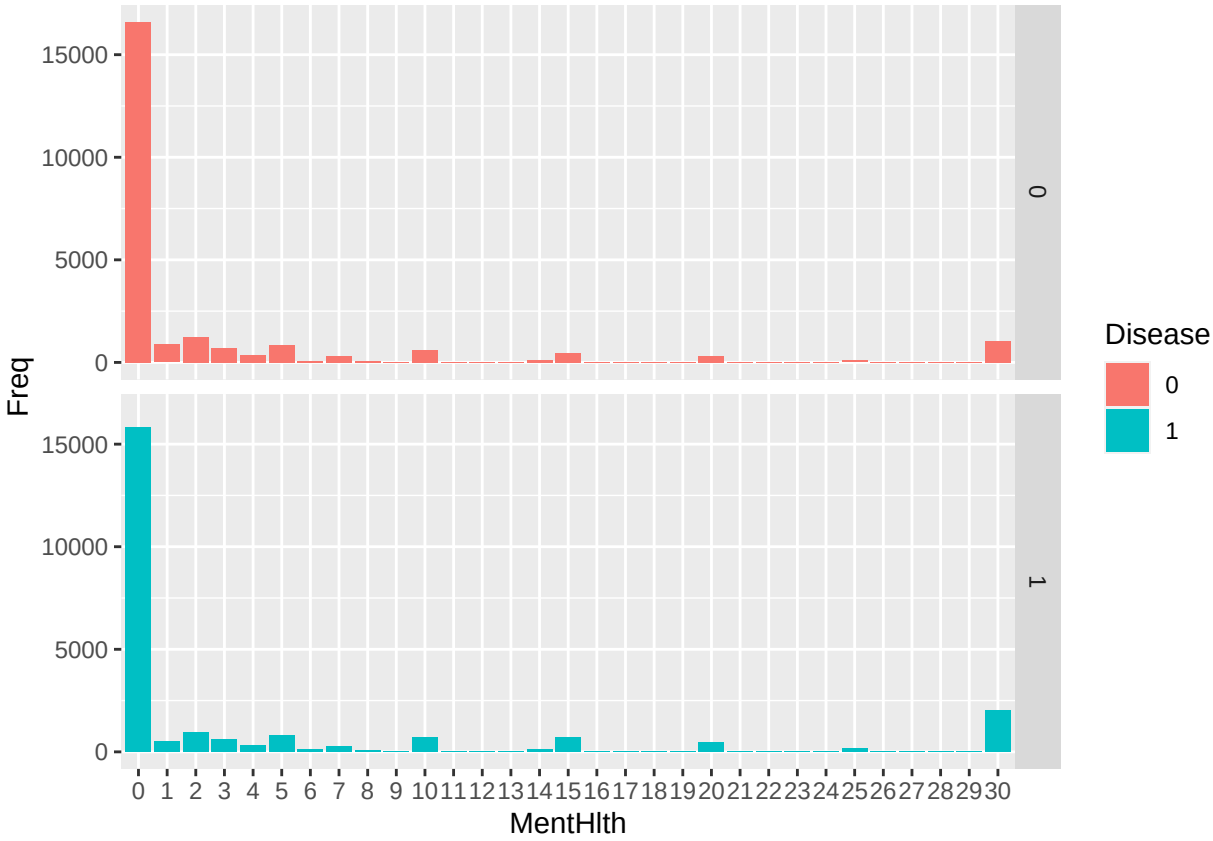


The **Diabetes** variable plot shows the group 2 with disease has a 32.97% of cases compared to 11.76% for the healthy in the same group.

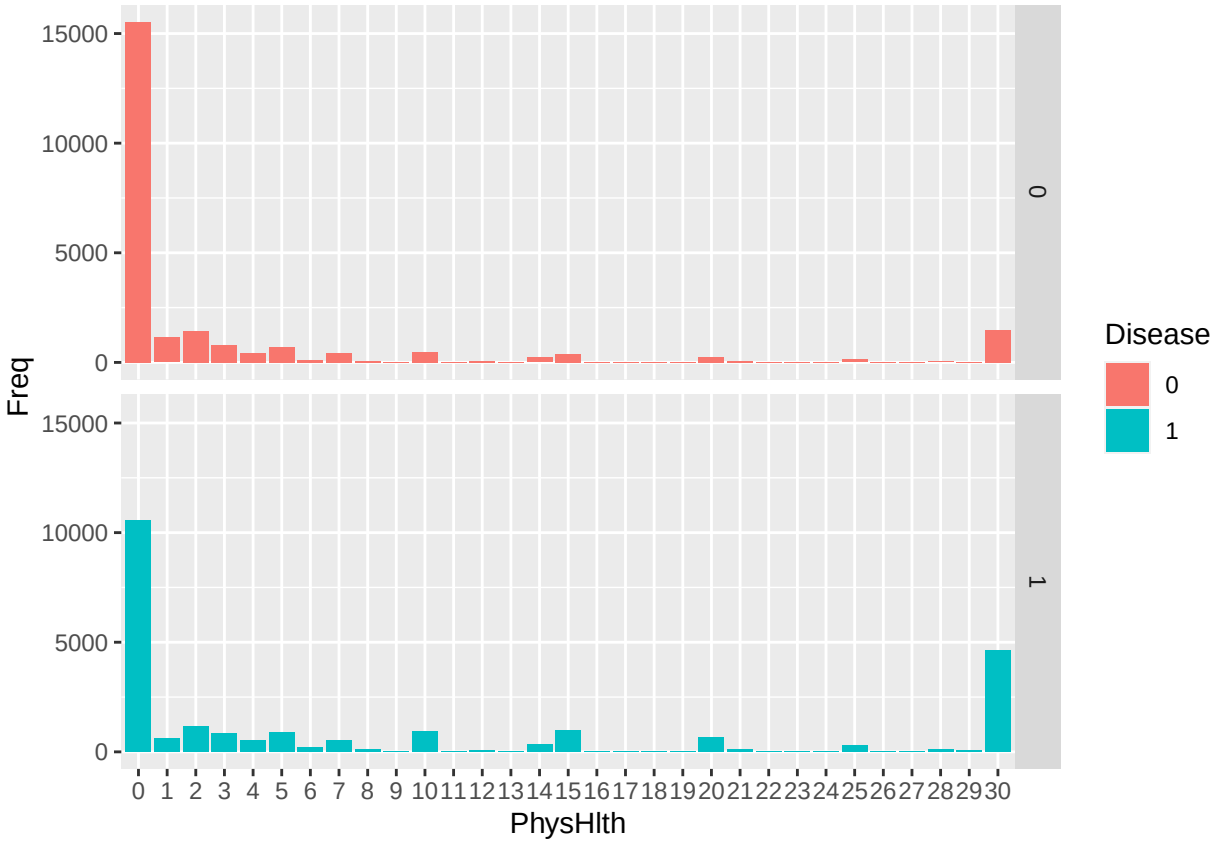
In the **GenHlth** variable we observe that groups 3, 4 and 5 have in general more disease cases and in groups 1 and 2 the healthy cases are more predominant.

We have a right skewed distribution for disease in the groups of the **Age** variable plot, in contrast to a more centered distribution for the healthy cases.

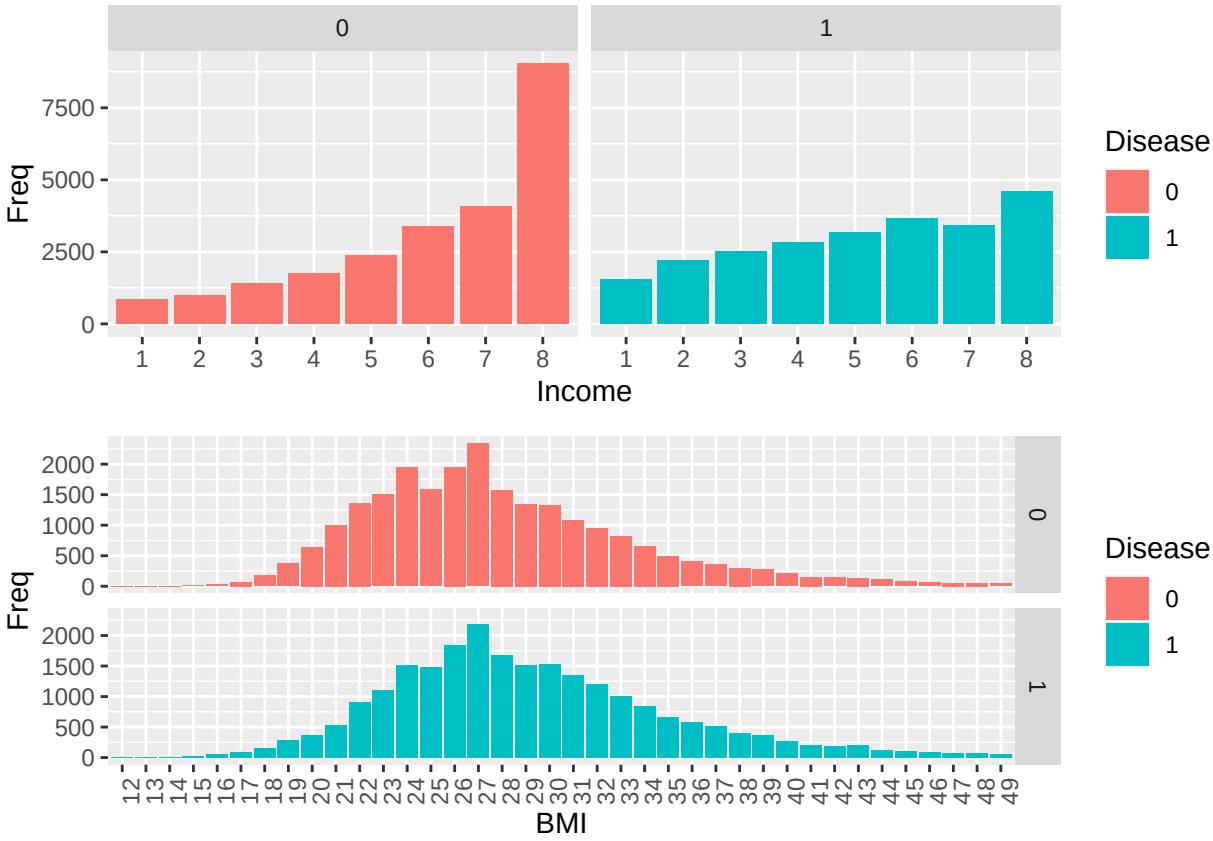
In the **Education** variable plot we see more disease in groups 2 through 5, but in group the group 6 the trend reverses



In the mental health plot the data is scattered among the 31 groups, but a big concentration occurs in the group 0 with 15804 disease cases vs 16611 healthy



We also have scattered data in the **PhysHlth** variable with big concentrations in the groups 0 and 30



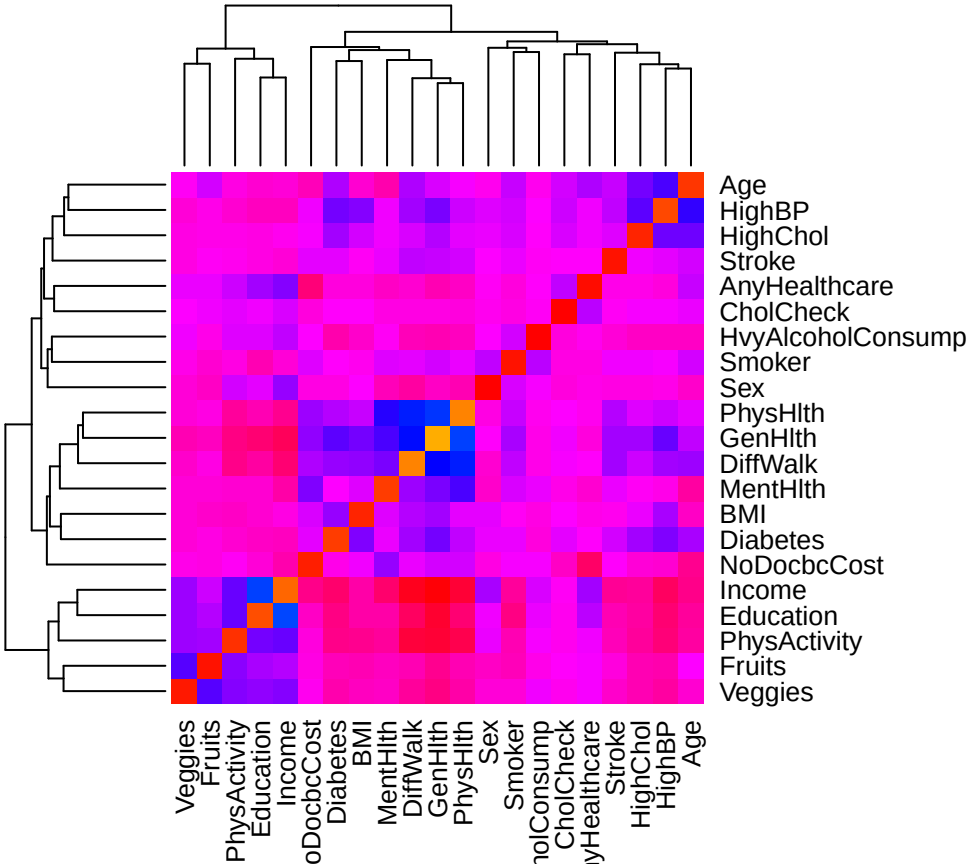
A slightly bigger count of disease cases in the **Income** variable groups 1 through 6, and the tendency reverses on the groups 7 and 8.

The **BMI** variable has a distribution centered around the 27 value with a long right tail, this plot doesn't look very informative from this perspective.

2.3 Heatmap

In this data set we count with 21 predictors, we start by scaling the dataset to plot the heatmap and perform a Principal Component Analysis.

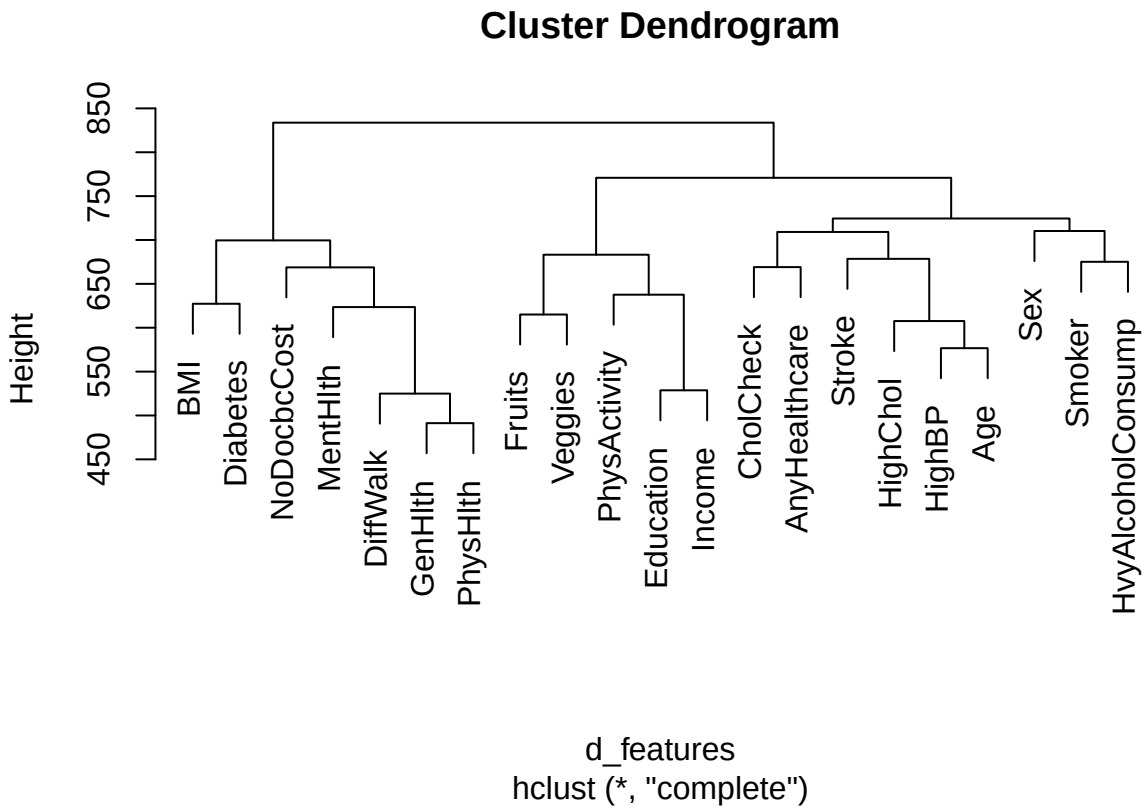
```
# Saving predictors of full data set into variable x
x <- as.matrix(hd[, 2:22])
# Subtract the column means of x
x_centered <- sweep(x, 2, colMeans(x))
# Divide by the column standard deviations
x_scaled <- sweep(x_centered, 2, colSds(x), FUN = "/")
# Calculate the distance between all samples
d_features <- dist(t(x_scaled))
# Plot heatmap
heatmap(as.matrix(d_features), scale = "column", col = rainbow(256))
```



If we look closely we find a few clusters with relevant correlation between predictors, but other than that we can't determine the variables that could drive an effective predictive algorithm.

2.4 Dendogram

Performing hierarchical clustering on the 21 features

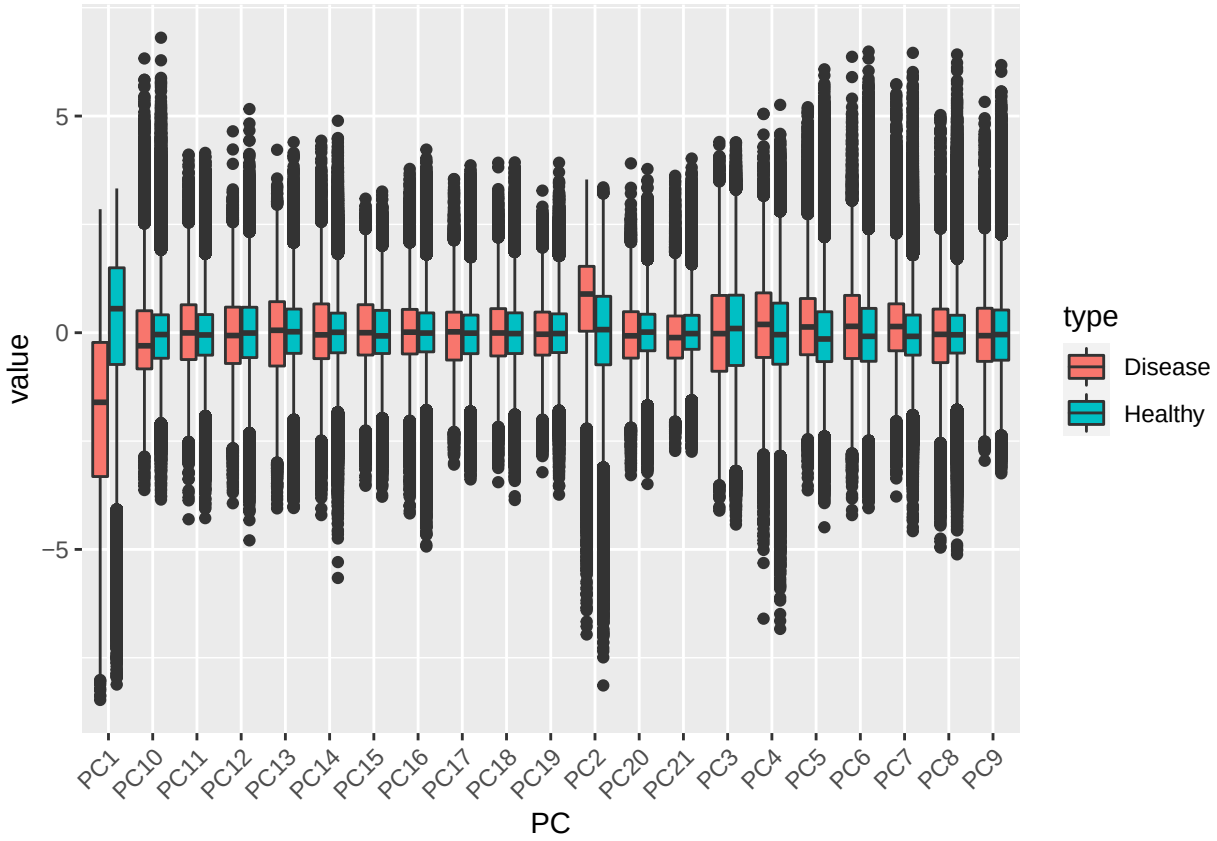


2.5 Principal Components

Performing Principal Component Analysis calculation on the full data set

```
pca <- prcomp(x_scaled)
extr <- summary(pca)

# Boxplot of PCA
data.frame(type = ifelse(hd[,1] == 0, "Healthy", "Disease"), pca$x) %>%
gather(key = "PC", value = "value", -type) %>%
ggplot(aes(PC, value, fill = type)) + geom_boxplot() +
theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



We have significant overlap in the interquartile ranges of the principal components, with a considerable amount of outliers, so we don't have a component different enough to differentiate between healthy cases and the ones with disease

Display Principal Components Values:

```
##                             PC1     PC2      PC3      PC4      PC5     PC6
## Standard deviation     1.883206 1.32447 1.167374 1.090573 1.071839 1.05807
## Proportion of Variance 0.168880 0.08353 0.064890 0.056640 0.054710 0.05331
## Cumulative Proportion  0.168880 0.25241 0.317310 0.373940 0.428650 0.48196

##                              PC7       PC8       PC9      PC10       PC11
## Standard deviation     0.9925322 0.9726872 0.9612985 0.9392518 0.9043794
## Proportion of Variance 0.0469100 0.0450500 0.0440000 0.0420100 0.0389500
## Cumulative Proportion  0.5288700 0.5739200 0.6179300 0.6599400 0.6988800

##                             PC12      PC13      PC14      PC15      PC16
## Standard deviation     0.888059 0.8608858 0.8539277 0.8461141 0.8382129
## Proportion of Variance 0.037550 0.0352900 0.0347200 0.0340900 0.0334600
## Cumulative Proportion  0.736440 0.7717300 0.8064500 0.8405500 0.8740000

##                             PC17      PC18      PC19      PC20      PC21
## Standard deviation     0.8154566 0.7465673 0.7145076 0.7014932 0.6488425
## Proportion of Variance 0.0316700 0.0265400 0.0243100 0.0234300 0.0200500
## Cumulative Proportion  0.9056700 0.9322100 0.9565200 0.9799500 1.0000000
```

The cumulative variance proportion reaches more than 90% until the component 17, we could use only those 17 predictors for our model but since the data set dimensions aren't cumbersome we are using the 21 predictors anyways

3 Models

Split data set into training and testing. we will split the full data set instead of the balanced version we created because the accuracy of the predictions depends heavily on the amount of samples we use for the training of our models.

```
#Spit data set into training and testing sets

set.seed(9, sample.kind = "Rounding")

test_index <- createDataPartition(y = hd$HeartDiseaseorAttack,
times = 1, p = 0.2, list = FALSE)

# Extract training set from full data set
train_set <- hd[-test_index, ]
# Extranct test set from full data set
test_set <- hd[test_index, ]

# Predictors for train set
train_x <- train_set[, 2:22]
# Outcomes for train_set
train_y <- train_set[,1]
# Predictors for test set
test_x <- test_set[, 2:22]
# Outcomes for test_set
test_y <- test_set[,1]
```

3.1 Random Forest Model:

For a first approach we use the random forest implementation in the Rborist package, because is faster than the one in the randomForest package.

Random forests are frequently used because they generate reasonable predictions across a wide range of data while requiring little configuration.

```
set.seed(9, sample.kind = "Rounding")
# We will use only 5-fold cross validation
control <- trainControl(method="cv", number = 5, p = 1)

grid <- expand.grid(minNode = c(1) , predFixed = c(10, 13, 16, 21))

train_rf <- train(train_x, as.factor(train_y), method = "Rborist",
nTree = 50, trControl = control, tuneGrid = grid, nSamp = 10000)

# Calculate the confusion matrix
cm <- confusionMatrix(predict(train_rf, test_x), as.factor(test_y))

# Show accuracy of the model
rf_acc <- cm$overall["Accuracy"]

models <- tibble(Method = "Random Forest", Accuracy = rf_acc)

models %>% knitr::kable()
```

Method	Accuracy
Random Forest	0.9074227

3.2 Extreme Gradient Boosting Model

Xgboost (extreme gradient boosting) is an advanced version of the gradient descent boosting technique, which is used for increasing the speed and efficiency of computation of the algorithm.

```
# xgboost
# Define training and testing sets
xgb_train <- xgb.DMatrix(data = as.matrix(train_x), label = train_y)
xgb_test <- xgb.DMatrix(data = as.matrix(test_x), label = test_y)

# Define final model
model_xgboost <- xgboost(data = xgb_train, max.depth = 3, nrounds = 86, verbose = 0)

# Use model to make predictions on test data
pred_y <- predict(model_xgboost, xgb_test)
```

Transform the regression in a binary classification:

The only thing that XGBoost does is a regression. XGBoost is using label vector to build its regression model.

If we think about the meaning of a regression applied to our data, the numbers we get are probabilities that a value will be classified as 1. Therefore, we will set the rule that if this probability for a specific value is $>$ 0.5 then the observation is classified as 1 or 0 otherwise.

```
# Transform prediction to binary
prediction <- as.numeric(pred_y > 0.5)
# Calculate accuracy
xgb_acc <- mean(prediction == test_y)

models <- bind_rows(models, tibble(Method = "XGBoost", Accuracy = xgb_acc))

models[2,] %>% knitr::kable()
```

Method	Accuracy
XGBoost	0.9100047

4 Results

```
models %>% knitr::kable()
```

Method	Accuracy
Random Forest	0.9074227
XGBoost	0.9100047

Our Random Forest prediction algorithm from the Rborist pakage delivered an accuracy of 90.7422737 %, it reached its maximum accuracy by using the 21 predictors available in the data set.

```
summary(train_rf)
```

```
##             Length Class      Mode
## sampler      4     Sampler    list
## leaf         2     Leaf       list
## forest       4     Forest     list
## predMap     21     -none-     numeric
## signature    6     Signature  list
## training     4     -none-     list
## prediction   3     PredictCtg list
## validation   3     ValidCtg   list
## xNames      21     -none-     character
## problemType  1     -none-     character
## tuneValue    2     data.frame list
## obsLevels    2     -none-     character
## param        2     -none-     list
```

```
train_rf$results
```

```
##   minNode predFixed  Accuracy       Kappa   AccuracySD     KappaSD
## 1       1        10 0.9054271 0.000000000 0.0000128827 0.000000000
## 2       1        13 0.9055454 0.004935979 0.0001951766 0.005252180
## 3       1        16 0.9056686 0.028224266 0.0001801968 0.013026797
## 4       1        21 0.9058262 0.087094149 0.0003797859 0.004512475
```

Meanwhile the extreme gradient boosting algorithm delivered an accuracy of 91.000473 %, a slight improvement over the random forest prediction, but it finishes the computations in a very short amount of time.

```
summary(model_xgboost)
```

```
##                Length Class              Mode
## handle              1 xgb.Booster.handle externalptr
## raw            104289 -none-             raw
## niter               1 -none-             numeric
## evaluation_log      2 data.table         list
## call               14 -none-             call
```

```
## params              2 -none-           list
## callbacks           1 -none-           list
## feature_names      21 -none-           character
## nfeatures           1 -none-           numeric
```

5 Conclusion

We obtained an accuracy above 90% with both algorithms which according to the machine learning literature is within an acceptable range and with very little tuning of parameters, we consider that the goal of the report was achieved for now, but keeping in mind that there is always room for improvement, each algorithm library has many parameters that can be tweaked to improve performance, as we learn more about them the more we will try to implement these improvements in the future and to explore alternative methods of predictions.

6 Reference

[1] “Centers for Disease Control and Prevention”, 2022, https://www.cdc.gov/heartdisease/index.htm

[2] Alex Teboul “Heart Disease Health Indicators Dataset”, 2022, https://www.kaggle.com/datasets/alexteboul/heart-disease-health-indicators-dataset

[3] Shanthi M, Pekka P, Norrving B, “Global Atlas on Cardiovascular Disease Prevention and Control”, 2011

[4] MedlinePlus, 2022, https://medlineplus.gov/heartdiseases.html

[5] Tunstall-Pedoe H. “Cardiovascular Risk and Risk Scores: ASSIGN, Framingham, QRISK and others: how to choose”, 2011