ML_Classificação

Leonardo Ribeiro

08/10/2021

Machine Learning - prevendo se um paciente tem doença cardíaca

O objetivo deste trabalho é criar um modelo de Machine Learning (ML) que aponte com razoável precisão se um paciente tem ou não uma doença cardíaca. Para isso, será usado um conjunto de dados retirado do Kaggle, um site que provê dados sobre diferentes tópicos e que frequentemente patrocina competições voltadas à Ciência de Dados.

Field	Description
Age	age of the patient [years]
Sex	sex of the patient [M: Male, F: Female]
ChestPainType	chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic]
RestingBP	resting blood pressure [mm Hg]
Cholesterol	serum cholesterol [mm/dl]
FastingBS	fasting blood sugar [1: if FastingBS > 120 mg/dl, 0: otherwise]
RestingECG	resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria
MaxHR	maximum heart rate achieved [Numeric value between 60 and 202]
ExerciseAngina	exercise-induced angina [Y: Yes, N: No]
Oldpeak	oldpeak = ST [Numeric value measured in depression]
ST_Slope	the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping]
HeartDisease	output class [1: heart disease, 0: Normal]

Para mais detalhes sobre cada uma das variáveis abordadas ao longo deste texto, visite **este link**.

O Problema

Um paciente chega ao hospital para fazer uma bateria de exames. O modelo de Machine Learning deve prever se ele está ou não doente.

Carregando os Dados

O primeiro passo será carregar o arquivo com os dados de interesse:

```
file <- list.files(pattern = ".csv")
raw_data <- read.csv(file)
dim(raw_data)
colnames(raw_data)
```

O conjunto de dados possui 918 linhas e 12 colunas (estas representam características dos pacientes). A coluna HeartDisease indica se um paciente tem uma doença cardíaca (1) ou se está saudável (0). Dessa forma, o modelo de ML aqui criado terá como objetivo prever o valor desta coluna.

Análise Exploratória

As colunas RestingBP (frequência cardíaca em repouso) e Cholesterol tem a seguinte distribuição de dados:

```
summary(raw_data[, c("Cholesterol", "RestingBP")])
```

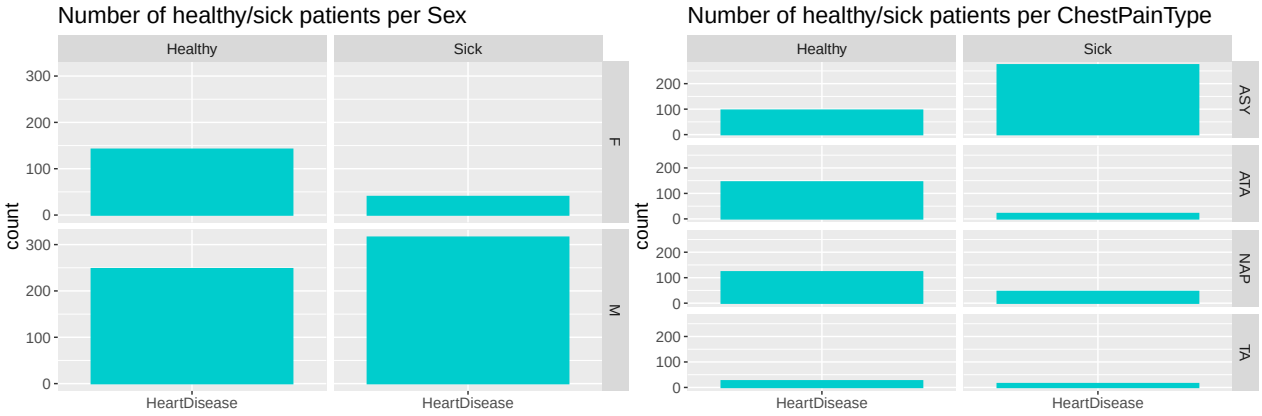
Nota-se que ambas as colunas tem o valor zero associado a elas. Isso indica que estes dois dados não foram coletados para todos os pacientes analisados.

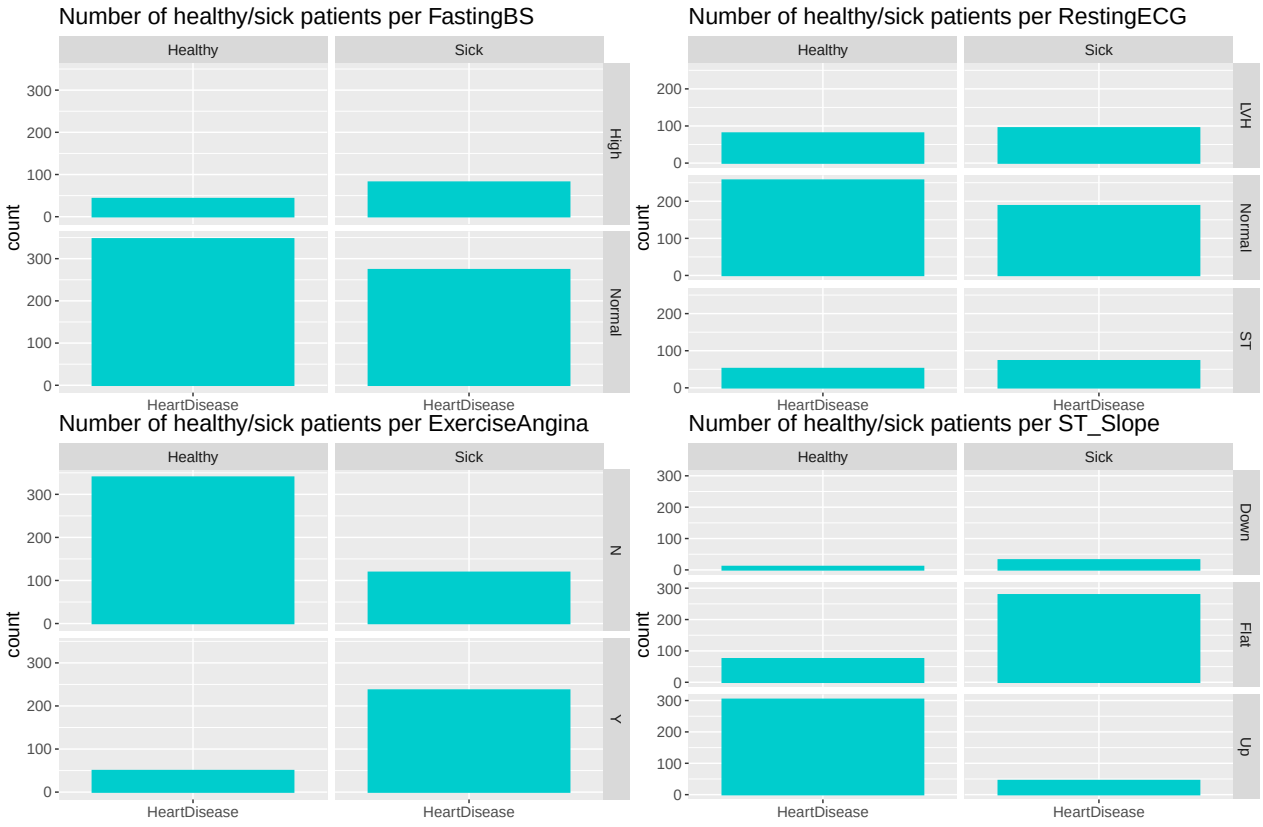
```
nrow(raw_data[raw_data$RestingBP == 0.0 ,])
nrow(raw_data[raw_data$Cholesterol == 0.0 ,])
raw_data <- raw_data[raw_data$RestingBP != 0.0,]
raw_data <- raw_data[raw_data$Cholesterol != 0.0,]
```

Há apenas um paciente para o qual a variável RestingBP não está presente, porém há 172 pacientes para os quais o nível de colestorol não foi medido. Com o intuito de evitar aberrações estatísticas, as linhas em que isto ocorre foram descartadas. Consequentemente, o número de linhas do dataset foi reduzido em quase 19%, o que fatalmente afetará a acurácia do modelo de ML.

O próximo passo é avaliar como as variáveis ChestPainType, FastingBS, RestingECG, ExerciseAngina e ST_Slope influenciam no fato de um paciente ter ou não uma doença cardíaca.

```
library(ggplot2)
raw_data$HeartDisease = sapply(raw_data$HeartDisease, function(x) {
  ifelse(x == '0', "Healthy", "Sick")
})
raw_data$FastingBS = sapply(raw_data$FastingBS, function(x) {
  ifelse(x == '0', "Normal", "High")
})
cols_of_interest <- c('Sex', 'ChestPainType', 'FastingBS', 'RestingECG',
                      'ExerciseAngina', 'ST_Slope')
dv.plot_multiple_bars_II(raw_data, cols_of_interest, 'HeartDisease', title = 'Number of healthy/sick pa
```





O primeiro fato que pode ser notado é que a maioria das mulheres que buscam o hospital para fazer a bateria de exames considerada estão saudáveis, enquanto o oposto é verdade para homens. Isso indica um maior cuidado pessoal por parte delas do que por parte deles.

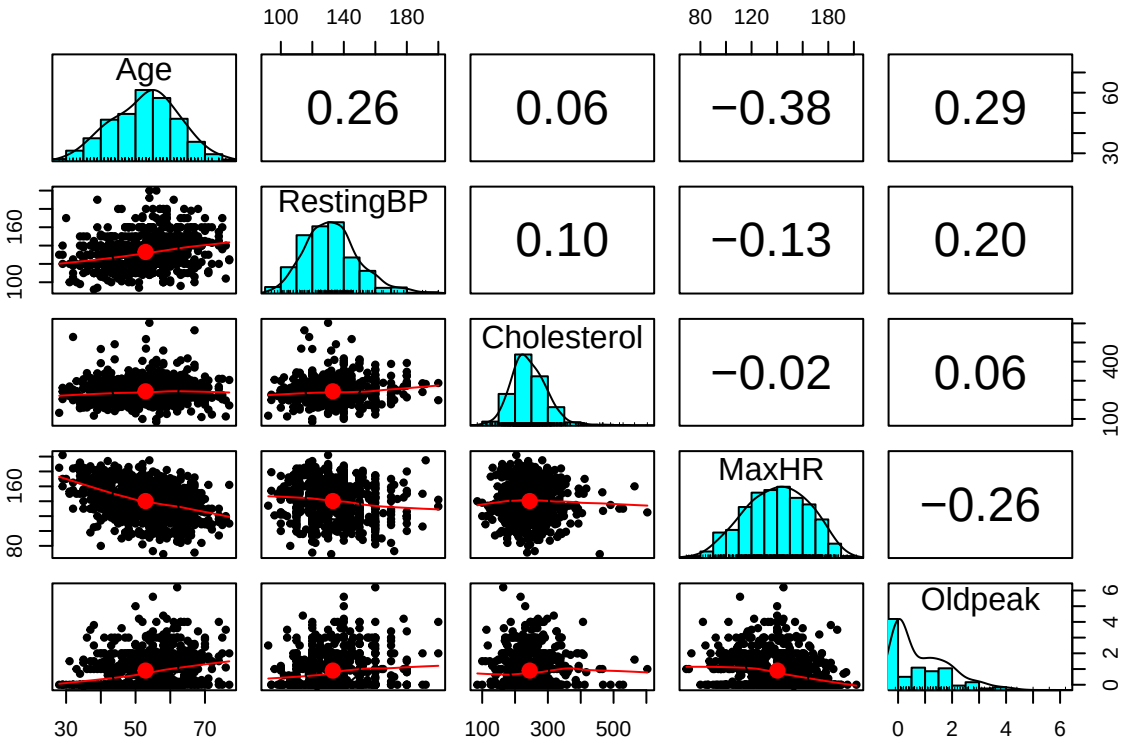
Outro ponto interessante é que a maioria das pessoas assintomáticas que realizam os exames em questão tem, em sua maioria, alguma doença cardíaca. Isso é um indicador de que esses exames são realizados para o acompanhamento rotineiro dessas doenças. Além disso, a maior parte dos pacientes com alta concentração de açúcar no sangue estão doentes, enquanto a maioria dos pacientes com concentrações normais estão saudáveis.

A variável RestingECG também parece influenciar na saúde dos pacientes. A maior parte eletrocardiogramas com resultados normais pertencem a pacientes saudáveis, enquanto o oposto é válido para pessoas com resultados aberrantes. Ainda, assim como RestingECG, a existência de angina provocada por exercícios (ExerciseAngina) parece uma condição importante para determinar se um paciente está ou não doente.

Por fim, outro indicador relevante é o ST_Slope. Pacientes com valor 'Up' para essa variável estão, em sua maioria, saudáveis. O oposto também parece ser verdade: pacientes para os quais essa variável tem valor 'Down' ou 'Flat' estão, na maioria das vezes, doentes.

As variáveis analisadas até então tem um caráter qualitativo, porém o dataset em questão também tem variáveis numéricas. Estas relacionam-se da seguinte maneira:

```
library(psych)
pairs.panels(raw_data[c('Age', 'RestingBP', 'Cholesterol', 'MaxHR', 'Oldpeak')])
```



Apesar de não haver nenhuma correlação forte, a variável Age tem uma correlação fraca com múltiplas colunas do dataset (quase média com MaxHR). Dessa forma, seria justificável a remoção de Age do processo de construção do modelo de ML. De fato, a correlação entre idade e MaxHR é conhecida desde a década de 80 (LONDEREE and MOESCHBERGER (1982) Effect of age and other factors on HRmax. Research Quarterly for Exercise & Sport, 53 (4), p. 297-304). Por outro lado, chama atenção como o colesterol não está correlacionado com nenhuma das outras variáveis apontadas acima.

Manipulação de Dados

Admitindo que variações pequenas na coluna Age são irrelevantes, torna-se interessante dividí-la em faixas:

```
raw_data <- dm.range_divide(raw_data, 'Age', 5, 1)
levels(raw_data$Age_range) <- c('28-37', '38-47', '48-57', '58-67', '68-77')
```

Além da coluna Age, as colunas RestingBP, Cholesterol, MaxHR e Oldpeak também são numéricas. Desta forma, o procedimento de fatorização também será repetido para elas.

```
raw_data <- dm.range_divide(raw_data, 'RestingBP', 5, 1)

raw_data <- dm.range_divide(raw_data, 'MaxHR', 5, 1)

raw_data <- dm.range_divide(raw_data, 'Oldpeak', 5, .1)

raw_data <- dm.range_divide(raw_data, 'Cholesterol', 5, 1)
```

As demais colunas já são fatores, porém seu tipo ainda é numérico. Por esse motivo elas serão fatorizadas.

```
fac_cols <- c('Sex', 'ChestPainType', 'FastingBS', 'RestingECG',
              'ExerciseAngina', 'ST_Slope', 'HeartDisease')

raw_data = dm.factorize_cols(raw_data, fac_cols)
```

Até esse ponto, nenhuma coluna for removida do dataset: colunas foram acrescentadas e outras tiverem o

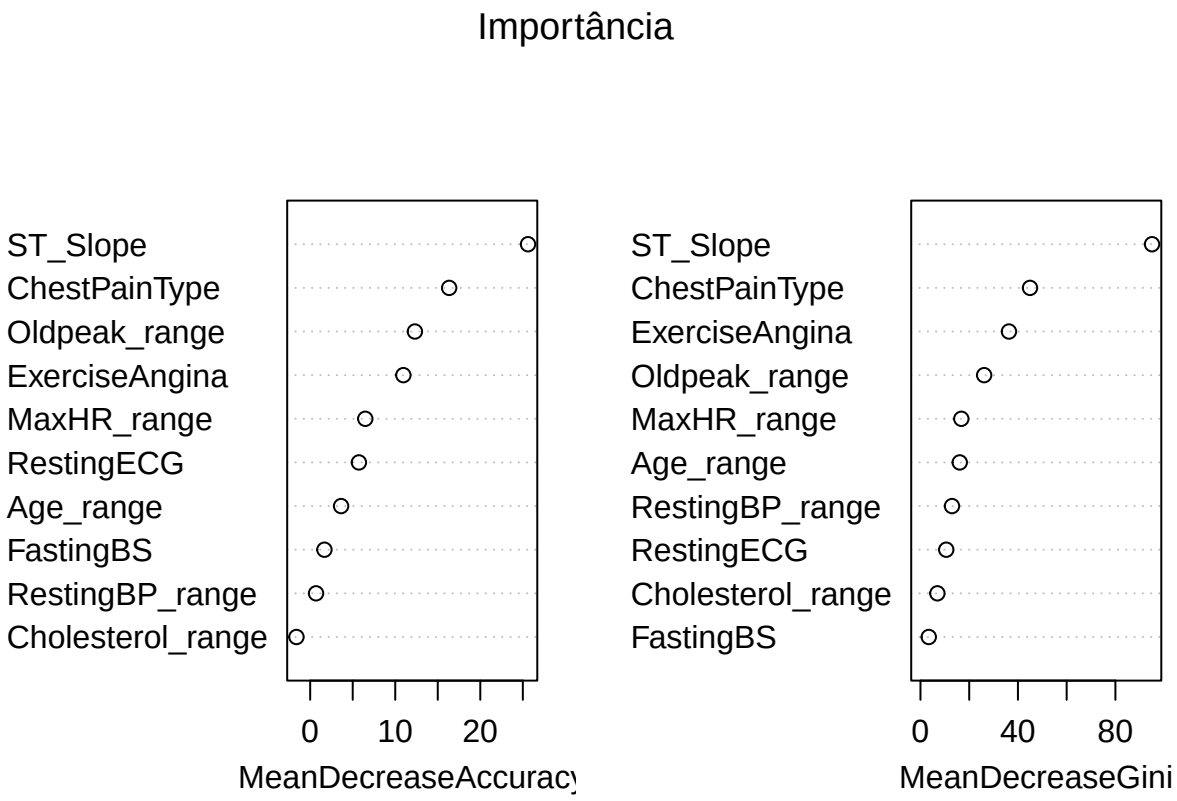
tipo de sua variável alterado. Para focar nos dados de interesse, as colunas Age, RestingBP, Cholesterol, MaxHR e Oldpeak serão descartadas. Além disso, como não queremos que o modelo diferencie pacientes pelo seu sexo, a coluna Sex também será removida.

```
cols_to_drop <- c('Sex', 'Age', 'RestingBP', 'Cholesterol', 'MaxHR', 'Oldpeak')
data <- dm.drop_cols(raw_data, cols_to_drop)
```

Contrução de modelo de ML

Usando randomForest, é possível encontrar a relevância que cada variável terá na tentativa de prever se um paciente está doente.

```
library(randomForest)
Importância <- randomForest(HeartDisease ~., data = data, ntree = 100,
                            nodesize = 10, importance = T)
varImpPlot(Importância)
```



Nota-se que, de acordo com ambos os critérios usados acima, as 5 variáveis mais importantes são as mesmas. Porém, como o dataset usado não tem um conjunto de variáveis muito grande, mesmo as que tem menos relevância serão mantidas.

O próximo passo é dividir o conjunto de dados em dados de treino e dados de teste.

```
random_indexes <- de.get_random_row_indexes(data, 80)
trainSet <- data[random_indexes, ]
testSet <- data[-random_indexes, ]
```

A divisão foi feita da seguinte maneira: 80% das linhas do dataset original foram separados para o treinamento do modelo, enquanto os 20% restantes serão usados para teste.

```
modelo <- randomForest(HeartDisease ~., data = trainSet, ntree = 100,
                       nodesize = 10)
modelo
```

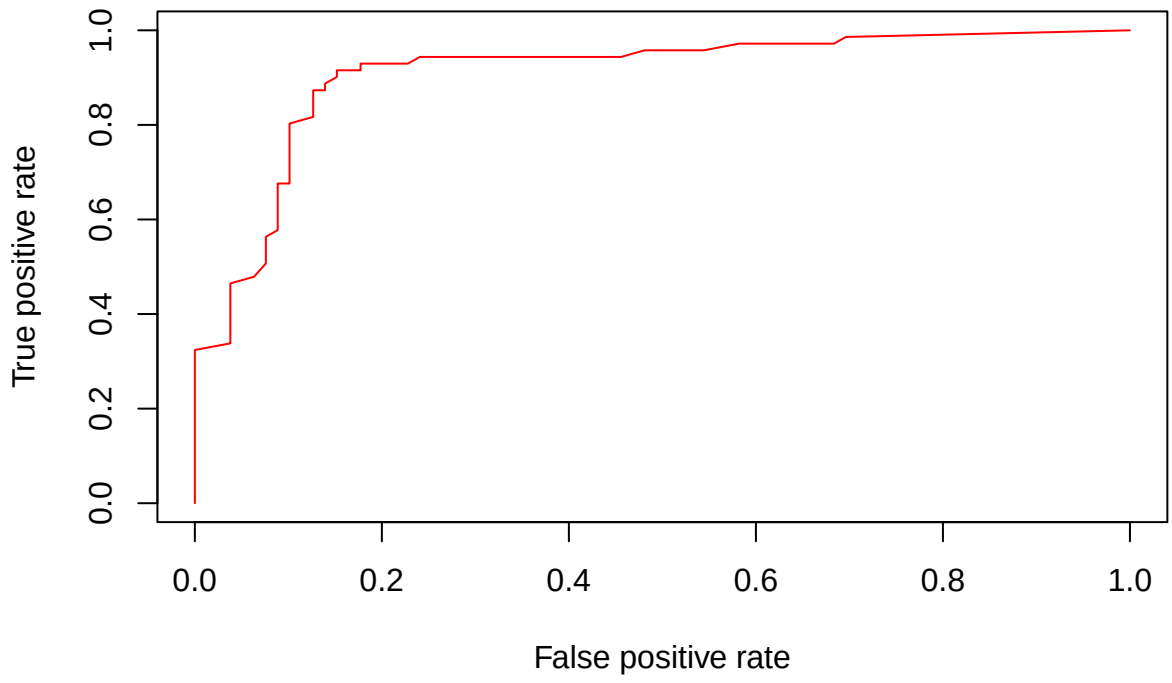
O modelo de ML criado tem, portanto, uma boa precisão: cerca de 85%. Em seguida, os dados de teste serão usados para verificar o poder de previsão do modelo:

```
library(caret)
previsao = predict(modelo, newdata = testSet)
previsoes = data.frame(observado = testSet$HeartDisease, previsto = previsao)
confusionMatrix(previsoes$observado, previsoes$previsto)
```

Usando o modelo de ML criado para prever o estado de um paciente, foi obtido um índice de acerto entre 85% e 90%. Esse modelo poderia ser otimizado e outras técnicas poderiam ser adotadas para obter uma precisão maior, porém isso foge do escopo deste texto.

Um outro indicativo da qualidade de um modelo de ML é a área abaixo da curva ROC. Ela simboliza a comparação entre diferentes indicadores e, portanto, é importante para que a precisão geral do modelo seja avaliada

```
library(ROCR)
class1 <- predict(modelo, newdata = testSet, type = 'prob')
class2 <- testSet$HeartDisease
pred <- prediction(class1[,2], class2)
perf <- performance(pred, 'tpr', 'fpr')
auc <- dv.get_auc(pred)
plot(perf, col = rainbow(10))
```




```
print(auc)
```

A área abaixo da curva está, portanto, acima de 0,9, o que indica que o modelo criado tem uma eficácia excelente.

Conclusão

O modelo de Machine Learning criado foi capaz de prever a existência de doenças cardíacas na grande maioria dos casos. Outros modelos poderiam ser criados, removendo as colunas Age e/ou colesterol, por exemplo, mas o valor encontrado para a precisão e para área abaixo da curva no modelo foram bons o suficiente para esta demonstração.