Atividade 1 - Aula de introdução ao Aprendizado de Máquina

Ricardo J. de Souza, Maíra A. H. do Nascimento, Hanna Diniz, Estevão Vargas, Lívia P. Coelho

02/05/2022

Conteúdo

Objetivo: Classificação usando Regressão Logística E Árvore de Decisão

Repositório

Documentação e códigos utilizados para criar este documento no link: introdução ao aprendizado de máquina.

Pacotes utilizados

Pacotes do R utilizados no trabalho:

```
# tidyverse
# patchwork
# caret
# corrplot
# corrgram
# tree
# xtable
# flextable
# smotefamily
# tree
```

```
library(tidyverse)

library(caret)
```

Carregar o banco para análise

Banco de dados utilizado: EXAMES-PNS-2013

```
banco <- read_csv("EXAMES-PNS-2013-selec.csv", locale = locale(decimal_mark = ","))
# tem que informar para o R o marcador de decimal do arquivo a ser importado.
```

Visualização dos nomes das variáveis, da dimensão e estrutura do banco de dados

Observamos que os nomes das variáveis estão em códigos, o banco de dados tem 8952 observações e 377 variáveis.

Diversas variáveis apresentam um número significativo de "NAs" e o tratamento será avaliado posteriormente.

```
names(banco)

glimpse(banco)
```

```
dim(banco)
```

```
## [1] 8952  377
```

Desfecho

Diagnóstico de Diabetes mellitus (DM)

Metodologia:

- Hemoglobina glicada : Diretrizes Sociedade Brasileira de Diabetes 2019-2020(FORTI et al., 2020)
- Diagnóstico prévio de DM

```
# Aspectos técnicos e laboratoriais de diagnóstico e acompanhamento
do diabetes mellitus.
 DM -> Hb glicada >= 6.5%.

# Consideramos como diabéticos as pessoas que tiveram diagnóstico
de diabetes por algum médico, incluindo diabetes gestacional,
(variável Q030 = 1 ou 2) e/ou hemoglobina glicada >= 6.5%.

# Z034 Hemoglobina Glicosilada (em %) + Q030 (diagnóstico de dibetes
por médico, incluindo gestacional)
```

Número de pacientes com a Hb glicada elevada

```
Ao todo, 595 pessoas tinham a Hb glicada acima de 6.5%
```

```
banco <- banco %>%
  mutate(hb_glic = if_else(Z034 >= 6.5, "Sim", "Não"))

banco %>%
  group_by(hb_glic) %>%
  summarise(N = n())
```

```
## # A tibble: 3 x 2
##   hb_glic     N
##   <chr>   <int>
## 1 Não      7946
## 2 Sim       595
## 3 <NA>      411
```

Número de pacientes com diagnóstico prévio de diabetes.

```
      Nessa população, 640 pessoas foram previamente diagnosticadas com DM.
```

```
banco <- banco %>%
  mutate(diabetes_prev = if_else(Q030 == 1 | Q030 == 2, "Sim", "Não"))

banco %>%
  group_by(diabetes_prev) %>%
  summarise(N = n())
```

```
## # A tibble: 3 x 2
##   diabetes_prev     N
##   <chr>         <int>
## 1 Não            7225
## 2 Sim             640
## 3 <NA>           1087
```

Relação entre pacientes com Hb glicada elevada e diagnóstico prévio de diabetes.

```
    Incluindo as duas condições anteriores, temos 923 pacientes com diabetes
    por uma e/ou outra condição.
```

```
banco %>%
  group_by(diabetes_prev, hb_glic) %>%
  summarise(N = n())
```

```
## # A tibble: 9 x 3
## # Groups:   diabetes_prev [3]
##   diabetes_prev hb_glic     N
##   <chr>         <chr>   <int>
## 1 Não           Não      6643
## 2 Não           Sim       260
## 3 Não           <NA>      322
## 4 Sim           Não       300
## 5 Sim           Sim       312
```

```
## 6 Sim           <NA>       28
## 7 <NA>          Não      1003
## 8 <NA>          Sim        23
## 9 <NA>          <NA>       61
```

```
banco <- banco %>%
  mutate(diabetes = if_else(Z034 >= 6.5 | Q030 == 1 | Q030 == 2 , "Sim", "Não"))
# criação da variável 'diabetes' de acordo com os parâmetros definidos.
```

Visualização da distribuição de diabetes na população de estudo

```
banco %>%
  group_by(diabetes) %>%
  summarise(N = n())
```

```
## # A tibble: 3 x 2
##   diabetes     N
##   <chr>    <int>
## 1 Não       6643
## 2 Sim        923
## 3 <NA>      1386
```

Variáveis preditoras

As variáveis preditoras foram selecionadas com base no artigo de Nascimento et. al.(NASCIMENTO et al., 2003)

dicionário de variáveis do banco de dados.

```
  Variáveis preditoras de acordo com o dicionário:

  # Z001    Sexo
  #         1   Masculino
  #         2   Feminino

  # Z002    Idade       Anos

  # IMC:
      imc = peso(kg)/altura(m)^2
      z004 = peso (kg); z005 = altura em cm.
  # imc = Z004 / ((Z005)/100)^2)

  # W00303 -> Circunferência da cintura - Final (em cm)

  # Z031    Colesterol Total (em mg/dL)
  # Z032    HDL Colesterol (em mg/dL)
  # Z033    LDL Colesterol (em mg/dL)

  # vldl = Z031- (Z032 + Z033) # segundo Nascimento et. al., o nível de vldl
  foi a variável com maior correlação com risco de diabetes

  # Z003    Cor ou raça
```

```
          #         1   Branca
          #         2   Preta
          #         3   Amarela
          #         4   Parda
          #         5   Indígena
          #         9   Ignorado

  # região -> Região do país

        # 1 Norte
        # 2 Nordeste
        # 3 Sudeste
        # 4 Sul
        # 5 Centro-Oeste

  # VDD004 ->   Nível de instrução mais elevado alcançado (pessoas de 5 anos
  ou mais de idade)
               #       1   Sem instrução
               #       2   Fundamental incompleto ou equivalente
               #       3   Fundamental completo ou equivalente
               #       4   Médio incompleto ou equivalente
               #       5   Médio completo ou equivalente
               #       6   Superior incompleto ou equivalente
               #       7   Superior completo
               #       .   Não aplicável
```

Banco de trabalho

Seleção de variáveis:

```
  Obs: criação de variável vldl e imc.
```

```
banco_trab <- banco %>%
  mutate(
    vldl = (Z031- (Z032+ Z033)), # VLDL = Colesterol total - (HDL + LDL)
    imc = round((z004 / ((z005)/100)^2), digits = 1)) %>%
  # arredonda o IMC com uma casa decimal
  select(diabetes, Z001, Z002,imc, W00303, Z031:Z033,vldl, Z003, VDD004, regiao)
```

Detalhes do banco de trabalho

```
glimpse(banco_trab)
```

```
## Rows: 8,952
## Columns: 12
## $ diabetes <chr> "Não", "Não", "Não", "Não", "Não", NA, "Não", "Não", "Não", "~
## $ Z001     <dbl> 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 1, 1, 1, 1, 1, 2, 2, 1~
## $ Z002     <dbl> 82, 46, 64, 65, 71, 83, 33, 41, 62, 80, 49, 53, 61, 34, 36, 3~
```

```
## $ imc      <dbl> 33.9, 20.8, 20.1, 23.2, 29.2, 25.3, 28.4, 30.2, 22.9, 21.7, 2~
## $ W00303   <dbl> 110.4, 77.3, 78.4, 96.5, 102.5, 97.0, 95.8, 104.0, 89.0, 97.0~
## $ Z031     <dbl> 156, 129, 158, 126, 220, 164, 226, 206, 172, 208, 107, 209, 2~
## $ Z032     <dbl> 27, 42, 37, 31, 40, 38, 36, 37, 44, 40, 38, 44, 68, 39, 46, 2~
## $ Z033     <dbl> 97, 72, 88, 58, 119, 93, 134, 133, 91, 133, 48, 130, 162, 99,~
## $ vldl     <dbl> 32, 15, 33, 37, 61, 33, 56, 36, 37, 35, 21, 35, 26, 51, 38, 3~
## $ Z003     <dbl> 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 1, 1, 1, 1, 1, 2, 2, 1~
## $ VDD004   <dbl> 1, 3, 1, 1, 1, 1, 3, 3, 3, 3, 3, 3, 3, 5, 5, 5, 5, 7, 3, 4, 3~
## $ regiao   <dbl> 1, 5, 1, 1, 1, 1, 5, 5, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 4~
```

Transformação das variáveis categóricas em fatores

```
  As variáveis que foram importadas como números foram recodificadas como fatores.
```

```
banco_trab$Z001 <- as.factor(banco_trab$Z001)
banco_trab$VDD004 <- as.factor(banco_trab$VDD004)
banco_trab$regiao <- as.factor(banco_trab$regiao)
```

Avaliação da quantidade de "NAs" e porcentagem por variável

Qual o máximo de NAs aceitáveis?(DZIURA et al., 2013)

```
  Dependendo do tipo de estudo, pode ser aceitável entre 5 a 25%.
  O ideal seria de 5% no máximo, mas são aceitáveis em alguns casos até 25%.
  Nós definimos que as variáveis deveriam ter menos do que 10% de "missing"
  para serem incluídas na análise.
```

```
banco_trab %>%
  summarise(N = n()) %>%
  tibble(Variaveis = colnames(banco_trab),
         Missing =  colSums(is.na(banco_trab)),
         Porcentagem = (colSums(is.na(banco_trab))*100/N)
  ) %>%
  select(Variaveis:Porcentagem) %>%
  arrange((Porcentagem)) %>%
  xtable::xtable() %>%
  flextable::as_flextable() %>%
  flextable::footnote(j = 2:4, part = 'header',
                      value = flextable::as_paragraph(
                        c(Variaveis = "Variáveis observadas;",
                          Missing = "Número de valores não observados;",
                          Porcentagem = "Relação entre os valores não observados e
                          o total de observações de cada variável (%).")),
                      ref_symbols = c("a", "b", "c")
                      )
```

	Variaveis[a]	Missing[b]	Porcentagem[c]
1	Z001	0.0	0.0
2	Z002	0.0	0.0
3	Z003	0.0	0.0
4	VDD004	0.0	0.0
5	regiao	6.0	0.1
6	imc	97.0	1.1
7	W00303	97.0	1.1
8	Z031	418.0	4.7
9	Z033	418.0	4.7
10	Z032	432.0	4.8
11	vldl	434.0	4.8
12	diabetes	1,386.0	15.5

[a]Variáveis observadas;

[b]Número de valores não observados;

[c]Relação entre os valores não observados e o total de observações de cada variável (%).

Análise exploratória

Após definirmos a variável desfecho e as possíveis preditoras, inicamos a análise exploratória.

```
summary(banco_trab)
```

```
##    diabetes          Z001          Z002             imc            W00303
##  Length:8952        1:3725   Min.   : 18.00   Min.   :13.10   Min.   : 50.00
##  Class :character   2:5227   1st Qu.: 34.00   1st Qu.:23.00   1st Qu.: 81.50
##  Mode  :character            Median : 45.00   Median :26.00   Median : 90.50
##                             Mean   : 46.84   Mean   :26.57   Mean   : 91.12
##                             3rd Qu.: 58.00   3rd Qu.:29.40   3rd Qu.: 99.90
##                             Max.   :104.00   Max.   :61.30   Max.   :149.70
##                                              NA's   :97      NA's   :97
##       Z031          Z032             Z033            vldl
##  Min.   : 67   Min.   :  6.00   Min.   : 22.0   Min.   :-25.00
##  1st Qu.:160   1st Qu.: 37.00   1st Qu.: 84.0   1st Qu.: 25.00
##  Median :184   Median : 45.00   Median :103.0   Median : 32.00
##  Mean   :187   Mean   : 45.98   Mean   :105.5   Mean   : 35.53
##  3rd Qu.:211   3rd Qu.: 53.00   3rd Qu.:124.0   3rd Qu.: 42.00
##  Max.   :433   Max.   :160.00   Max.   :261.0   Max.   :288.00
##  NA's   :418   NA's   :432      NA's   :418     NA's   :434
```

```
##       Z003       VDD004    regiao
##  Min.   :1.000   1:1685   1   :2297
##  1st Qu.:1.000   2:2293   2   :3053
##  Median :2.000   3: 941   3   :1514
##  Mean   :1.584   4: 443   4   :1075
##  3rd Qu.:2.000   5:2291   5   :1007
##  Max.   :2.000   6: 380   NA's:   6
##                  7: 919
```

Após a análise inicial, verificamos que as variáveis Z001(sexo) e Z003(raça) tinham seus preenchimentos idênticos. Como a raça estava com preenchimento binário (1,2), assumimos que houve erro no preenchimento dessa variável.

```
table(banco_trab$Z001, banco_trab$Z003)
```

```
##
##        1    2
##   1 3725    0
##   2    0 5227
```

Exclusão da variável raça (Z003).

```
banco_trab <- banco_trab %>% select(everything(), -Z003)
```

Lidando com NAs da variável de desfecho

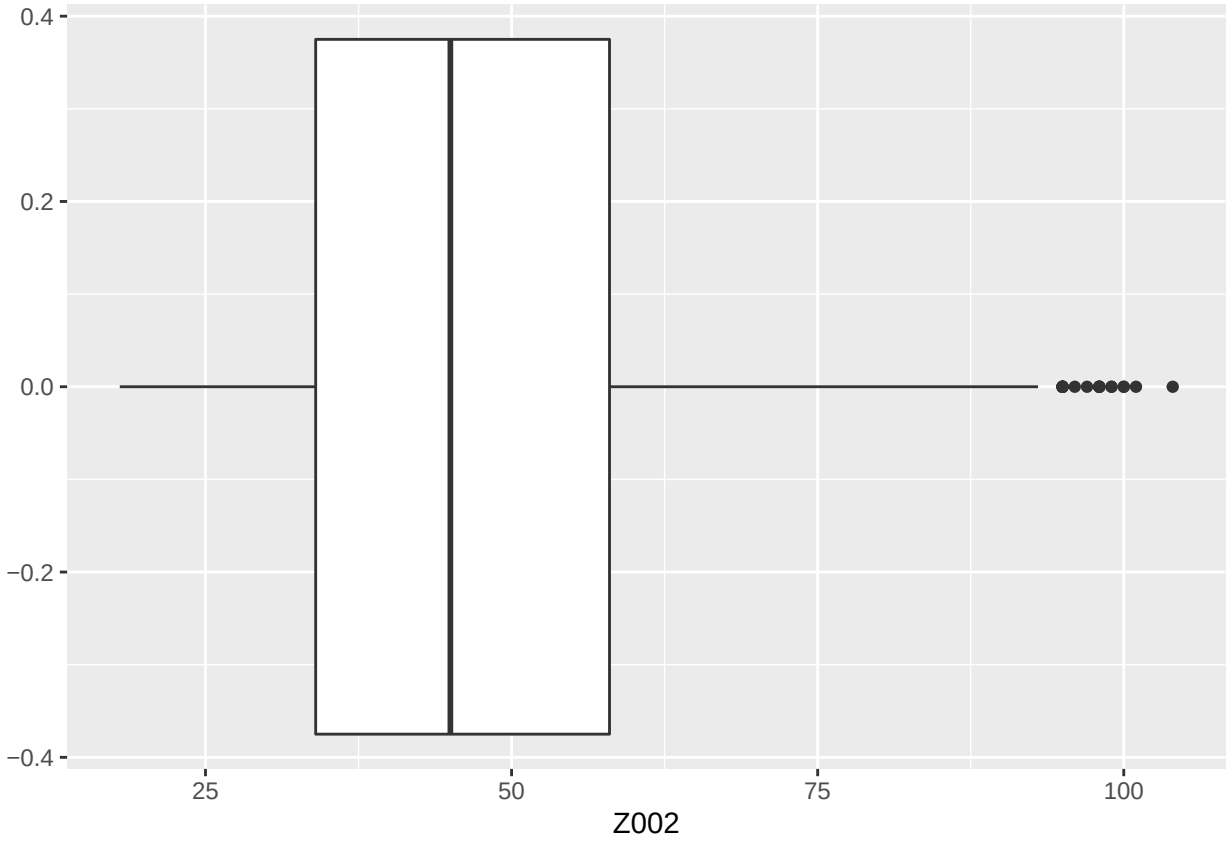
> Dúvida: não seria melhor primeiro imputar dados faltantes nas variáveis preditoras antes de eliminar grande parte do banco de dados? Perderemos mais de 1300 observações retirando os NAs do desfecho.
> Preferimos imputar os missings pelas médias (retirando-se os outliers) antes de retirar os faltantes da variável de desfecho.
>
> Talvez a melhor opção seja aplicar a mediana nas variáveis com muitos outliers.

Visualização dos dados

Testes de outliers Box plot:

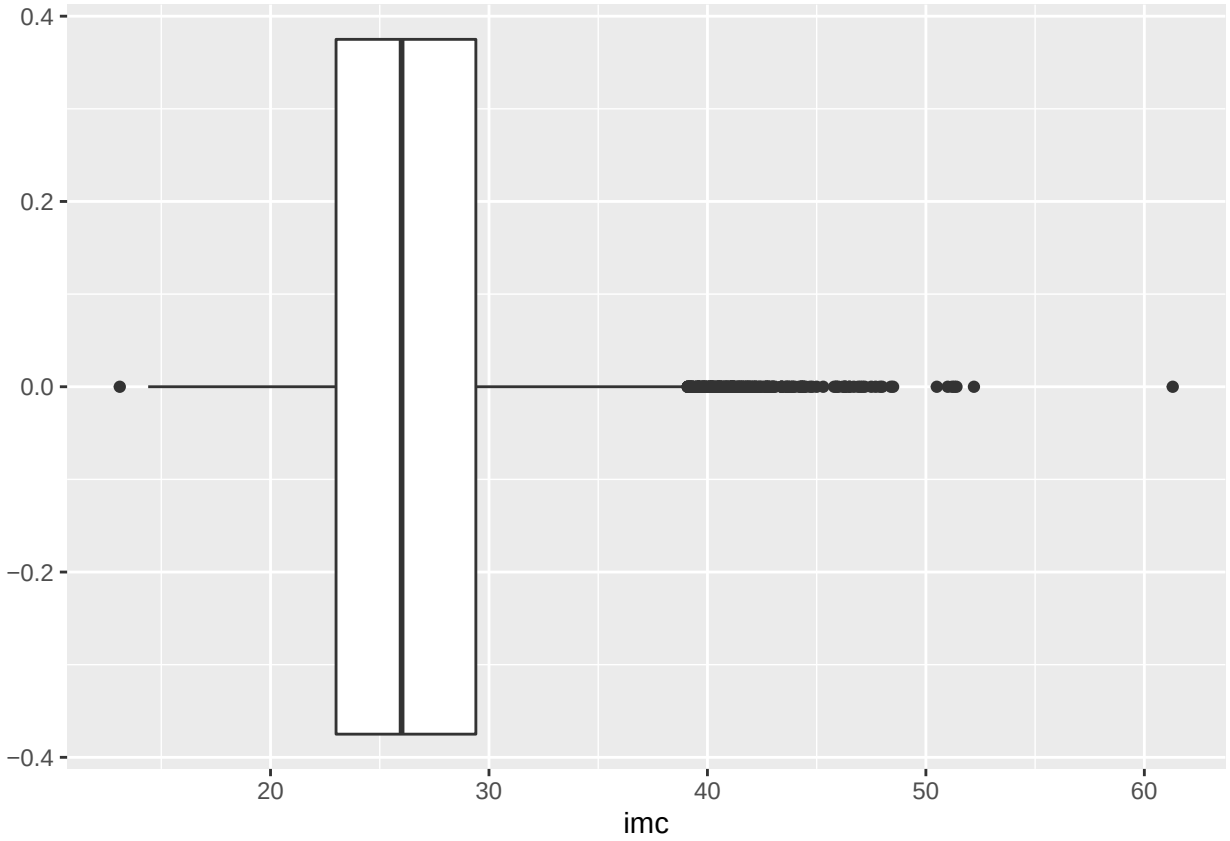
- **Idade**

```
banco_trab %>%
  ggplot(aes(Z002))+
  geom_boxplot()
```



- **IMC**

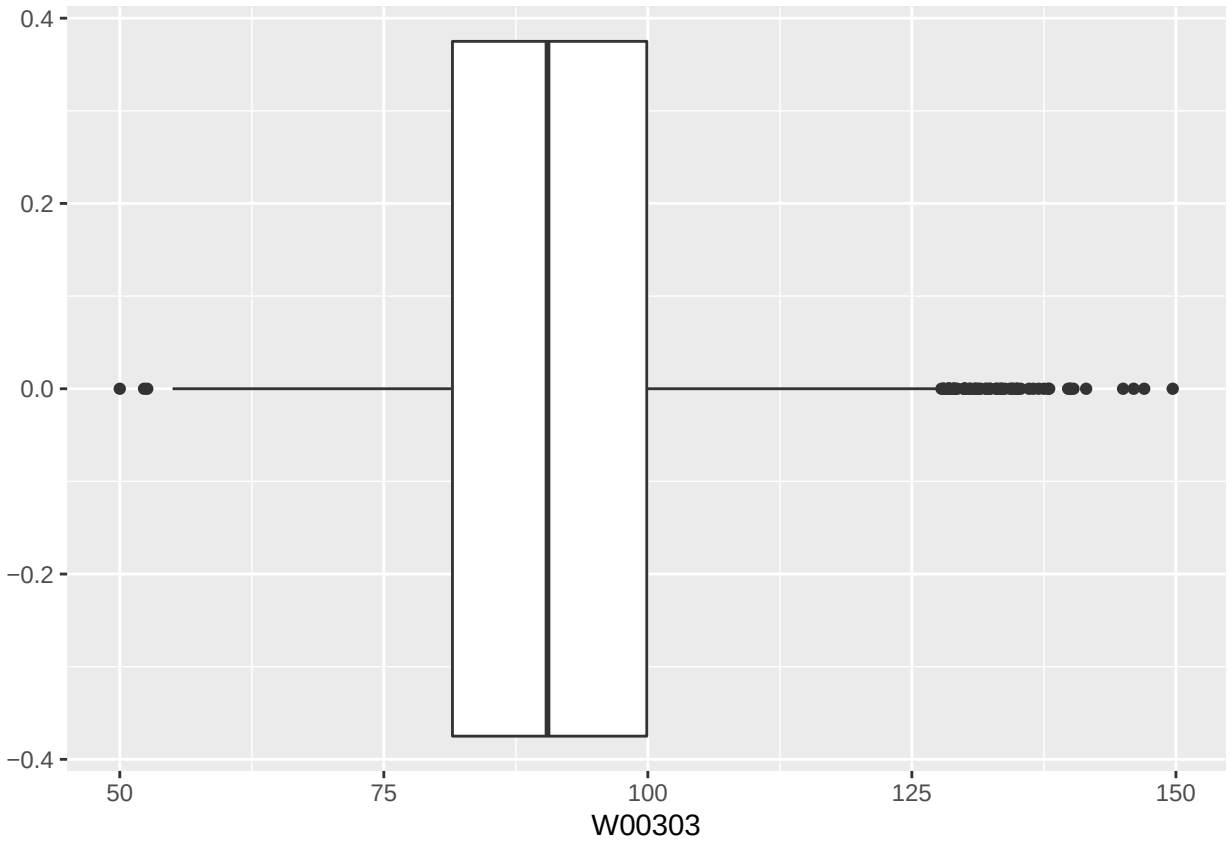
```
banco_trab %>%
  ggplot(aes(imc))+
  geom_boxplot()
```



- **Circunferência da cintura**

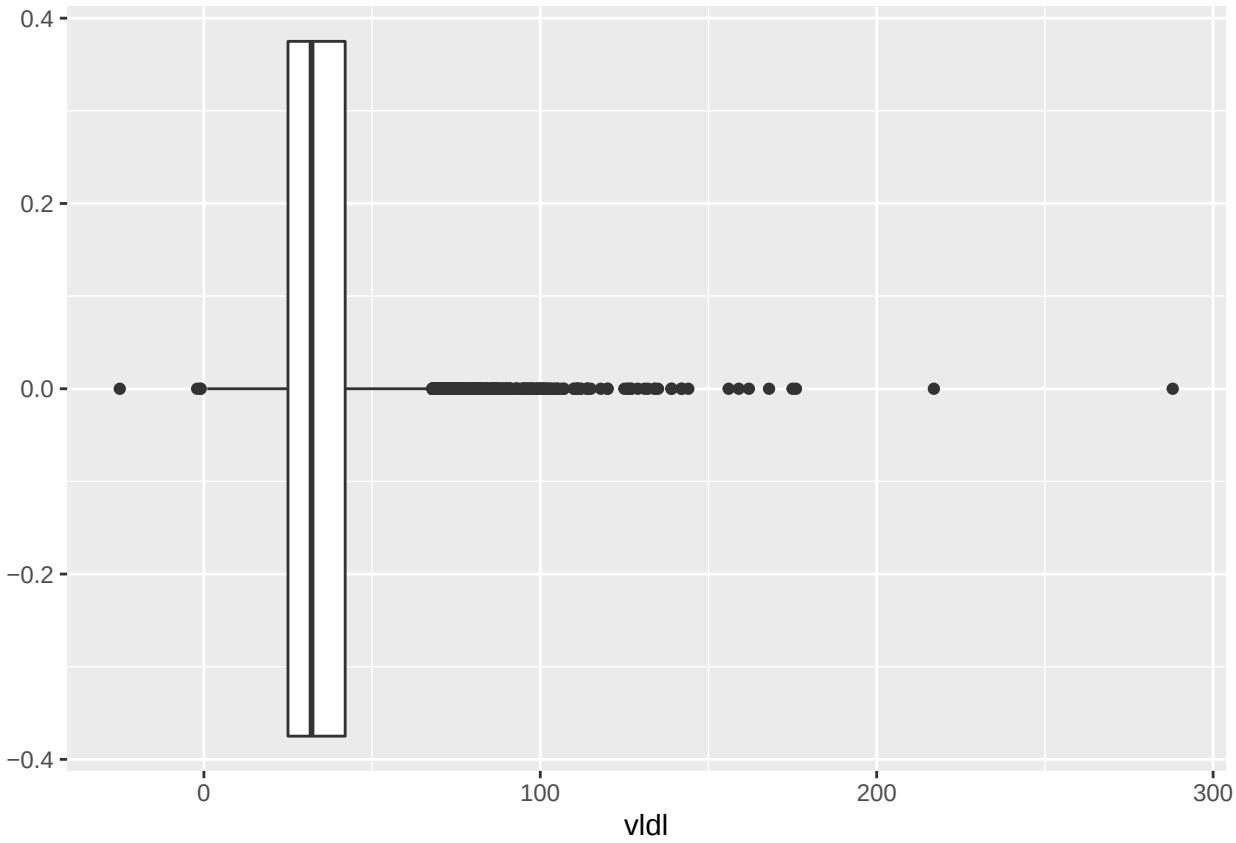
```
banco_trab %>%
  ggplot(aes(W00303))+
  geom_boxplot()
```

```
## Warning: Removed 97 rows containing non-finite values (stat_boxplot).
```



- **VLDL**

```
banco_trab %>%
  ggplot(aes(vldl))+
  geom_boxplot()
```



Teste outliers

Ausência de verdadeiros outliers em idade

```
test <- EnvStats::rosnerTest(banco_trab$Z002, k=8)

test$all.stats
```

```
##   i   Mean.i     SD.i Value Obs.Num    R.i+1 lambda.i+1 Outlier
## 1 0 46.84428 16.45230   104    7839 3.474026   4.539019   FALSE
## 2 1 46.83790 16.44212   101    3403 3.294106   4.538995   FALSE
## 3 2 46.83184 16.43307   100     810 3.235437   4.538971   FALSE
## 4 3 46.82590 16.42437   100    4108 3.237512   4.538947   FALSE
## 5 4 46.81996 16.41567    99    2809 3.178673   4.538924   FALSE
## 6 5 46.81413 16.40731    99    3692 3.180647   4.538900   FALSE
## 7 6 46.80829 16.39895    98    2114 3.121646   4.538876   FALSE
## 8 7 46.80257 16.39093    98    2745 3.123522   4.538852   FALSE
```

Valores de IMC acima de 50 foram considerados outliers

```
test <- EnvStats::rosnerTest(banco_trab$imc, k=20)

test$all.stats
```

```
##     i   Mean.i     SD.i Value Obs.Num    R.i+1 lambda.i+1 Outlier
## 1   0 26.57346 5.096999  61.3     171 6.813134   4.536699    TRUE
## 2   1 26.56954 5.083906  52.2    2849 5.041490   4.536674    TRUE
## 3   2 26.56664 5.076889  51.4     798 4.891451   4.536650    TRUE
## 4   3 26.56384 5.070309  51.3    3581 4.878630   4.536626    TRUE
## 5   4 26.56104 5.063772  51.2    5349 4.865731   4.536602    TRUE
## 6   5 26.55826 5.057280  51.0    1203 4.832982   4.536578    TRUE
## 7   6 26.55550 5.050885  50.5    8900 4.740655   4.536554    TRUE
## 8   7 26.55279 5.044751  48.5    3323 4.350504   4.536530   FALSE
## 9   8 26.55031 5.039636  48.4    7263 4.335569   4.536506   FALSE
## 10  9 26.54784 5.034563  48.0    7100 4.260978   4.536482   FALSE
## 11 10 26.54542 5.029677  47.9    8554 4.245717   4.536458   FALSE
## 12 11 26.54300 5.024832  47.7    4098 4.210489   4.536434   FALSE
## 13 12 26.54061 5.020076  47.5    2297 4.175115   4.536410   FALSE
## 14 13 26.53824 5.015408  47.2    2253 4.119657   4.536386   FALSE
## 15 14 26.53590 5.010874  47.1    1551 4.103894   4.536361   FALSE
## 16 15 26.53357 5.006381  47.0     420 4.088068   4.536337   FALSE
## 17 16 26.53126 5.001929  46.9    3074 4.072177   4.536313   FALSE
## 18 17 26.52895 4.997516  46.7    7172 4.036214   4.536289   FALSE
## 19 18 26.52667 4.993190  46.5     616 4.000114   4.536265   FALSE
## 20 19 26.52441 4.988948  46.5    7299 4.003968   4.536241   FALSE
```

Valores acima de 100mg/dl foram considerados outliers no test. Temos que avaliar a retirada do cálculo da média para imputação dos NAs ou utilizar a mediana. Mas vamos manter para avaliação dos modelos preditivos e retirar somente se melhorarem a predição.

```
  test <- EnvStats::rosnerTest(banco_trab$vldl, k=67)

test$all.stats
```

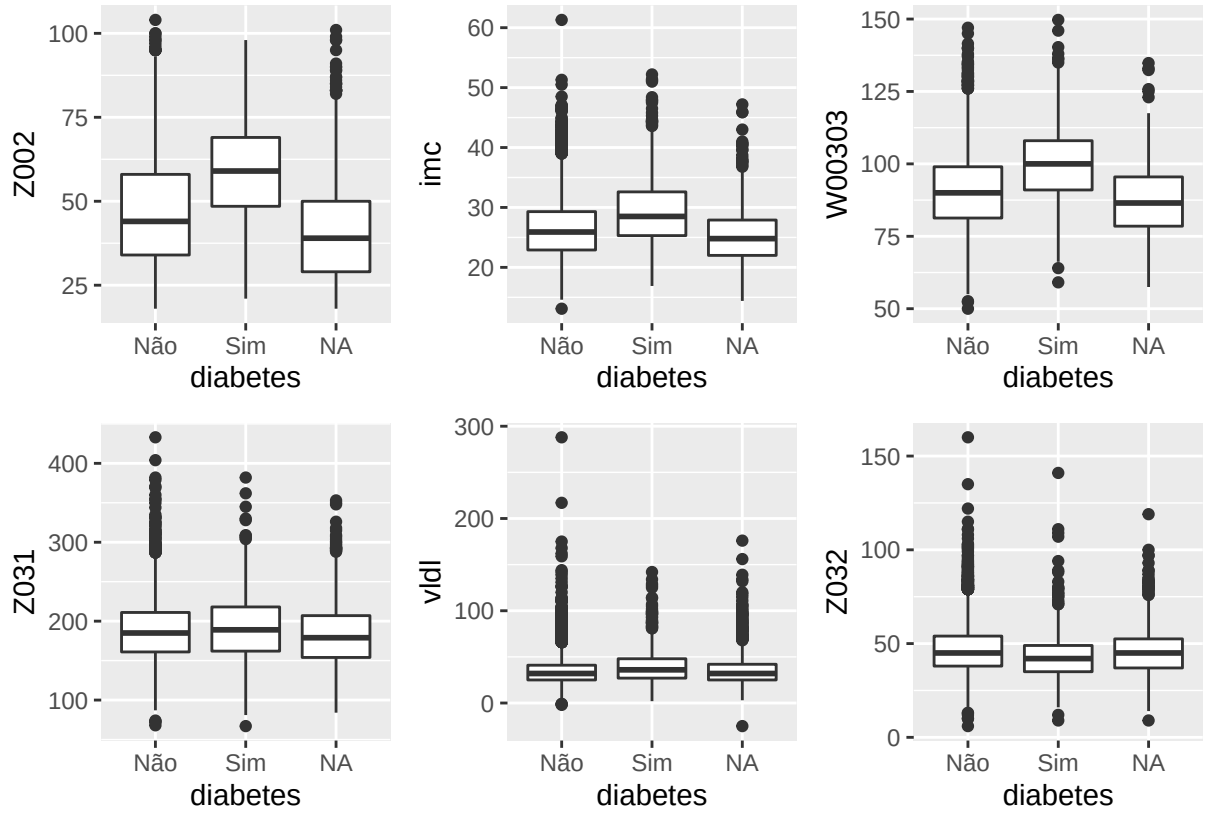
```
##     i   Mean.i     SD.i Value Obs.Num     R.i+1 lambda.i+1 Outlier
## 1   0 35.53135 16.48751   288    8925 15.312719   4.528425    TRUE
## 2   1 35.50170 16.25990   217    1972 11.162325   4.528400    TRUE
## 3   2 35.48039 16.14145   176    5622  8.705515   4.528374    TRUE
## 4   3 35.46389 16.07039   175     462  8.682808   4.528349    TRUE
## 5   4 35.44750 16.00001   168     599  8.284526   4.528324    TRUE
## 6   5 35.43193 15.93631   162      17  7.942118   4.528299    TRUE
## 7   6 35.41706 15.87808   159    8783  7.783242   4.528274    TRUE
## 8   7 35.40254 15.82239   156    5581  7.621948   4.528249    TRUE
## 9   8 35.38837 15.76922   144     529  6.887574   4.528224    TRUE
## 10  9 35.37560 15.72612   142    3003  6.780085   4.528199    TRUE
## 11 10 35.36307 15.68449   142    3625  6.798878   4.528174    TRUE
## 12 11 35.35053 15.64273   139    1660  6.626045   4.528149    TRUE
## 13 12 35.33835 15.60322   139    5575  6.643605   4.528124    TRUE
## 14 13 35.32616 15.56359   135    8168  6.404295   4.528099    TRUE
## 15 14 35.31444 15.52693   134     173  6.355770   4.528073    TRUE
## 16 15 35.30283 15.49091   134    3802  6.371297   4.528048    TRUE
## 17 16 35.29123 15.45478   132    6500  6.257530   4.528023    TRUE
## 18 17 35.27985 15.42005   131    7757  6.207511   4.527998    TRUE
## 19 18 35.26859 15.38596   129    2192  6.092009   4.527973    TRUE
## 20 19 35.25756 15.35323   127    3202  5.975448   4.527948    TRUE
## 21 20 35.24676 15.32184   127    3983  5.988395   4.527923    TRUE
```

```
## 22 21 35.23597 15.29037   126    3925  5.936026   4.527898    TRUE
## 23 22 35.22528 15.25952   126    6947  5.948725   4.527873    TRUE
## 24 23 35.21460 15.22860   125    6090  5.895841   4.527847    TRUE
## 25 24 35.20403 15.19830   120     147  5.579307   4.527822    TRUE
## 26 25 35.19404 15.17131   120    3002  5.589890   4.527797    TRUE
## 27 26 35.18406 15.14426   118     362  5.468470   4.527772    TRUE
## 28 27 35.17430 15.11846   115    3917  5.280016   4.527747    TRUE
## 29 28 35.16490 15.09450   114     902  5.222770   4.527722    TRUE
## 30 29 35.15561 15.07111   114    3709  5.231490   4.527697    TRUE
## 31 30 35.14632 15.04768   114    4448  5.240254   4.527671    TRUE
## 32 31 35.13703 15.02420   114    7226  5.249063   4.527646    TRUE
## 33 32 35.12774 15.00067   112    6526  5.124588   4.527621    TRUE
## 34 33 35.11868 14.97832   112    7808  5.132841   4.527596    TRUE
## 35 34 35.10962 14.95592   111     816  5.074270   4.527571    TRUE
## 36 35 35.10067 14.93408   111     905  5.082289   4.527546    TRUE
## 37 36 35.09172 14.91220   111    3905  5.090346   4.527520    TRUE
## 38 37 35.08277 14.89028   111    5842  5.098441   4.527495    TRUE
## 39 38 35.07382 14.86832   110     144  5.039318   4.527470    TRUE
## 40 39 35.06498 14.84691   110    7467  5.047180   4.527445    TRUE
## 41 40 35.05615 14.82546   107     814  4.852724   4.527420    TRUE
## 42 41 35.04766 14.80572   107    1745  4.859766   4.527395    TRUE
## 43 42 35.03917 14.78595   106    1954  4.799207   4.527369    TRUE
## 44 43 35.03080 14.76671   105     562  4.738306   4.527344    TRUE
## 45 44 35.02254 14.74801   105    1154  4.744876   4.527319    TRUE
## 46 45 35.01428 14.72927   105    4461  4.751474   4.527294    TRUE
## 47 46 35.00602 14.71049   104    5054  4.690120   4.527269    TRUE
## 48 47 34.99788 14.69225   104    8211  4.696499   4.527243    TRUE
## 49 48 34.98973 14.67397   103     552  4.634757   4.527218    TRUE
## 50 49 34.98170 14.65621   103     947  4.640921   4.527193    TRUE
## 51 50 34.97367 14.63842   103    2275  4.647109   4.527168    TRUE
## 52 51 34.96563 14.62060   102     595  4.584925   4.527143    TRUE
## 53 52 34.95771 14.60330   102     864  4.590900   4.527117    TRUE
## 54 53 34.94979 14.58597   102    2019  4.596898   4.527092    TRUE
## 55 54 34.94187 14.56861   102    5751  4.602920   4.527067    TRUE
## 56 55 34.93395 14.55122   101    1000  4.540243   4.527042    TRUE
## 57 56 34.92614 14.53434   101    1751  4.546052   4.527016    TRUE
## 58 57 34.91833 14.51743   101    2929  4.551884   4.526991    TRUE
## 59 58 34.91052 14.50050   101    5155  4.557738   4.526966    TRUE
## 60 59 34.90271 14.48354   101    8162  4.563615   4.526941    TRUE
## 61 60 34.89489 14.46655   100    1023  4.500390   4.526915    TRUE
## 62 61 34.88719 14.45007   100    3252  4.506056   4.526890    TRUE
## 63 62 34.87949 14.43356   100    3699  4.511743   4.526865    TRUE
## 64 63 34.87179 14.41702   100    3998  4.517451   4.526840    TRUE
## 65 64 34.86409 14.40046   100    8097  4.523182   4.526814    TRUE
## 66 65 34.85638 14.38387   100    8699  4.528934   4.526789    TRUE
## 67 66 34.84867 14.36726    99    2132  4.465106   4.526764   FALSE
```

Gráficos de caixa entre variável desfecho e variáveis numéricas

Observamos que os pacientes mais com diabetes são mais idosos em geral, tem o IMC, circunferência abdominal mais elevados.

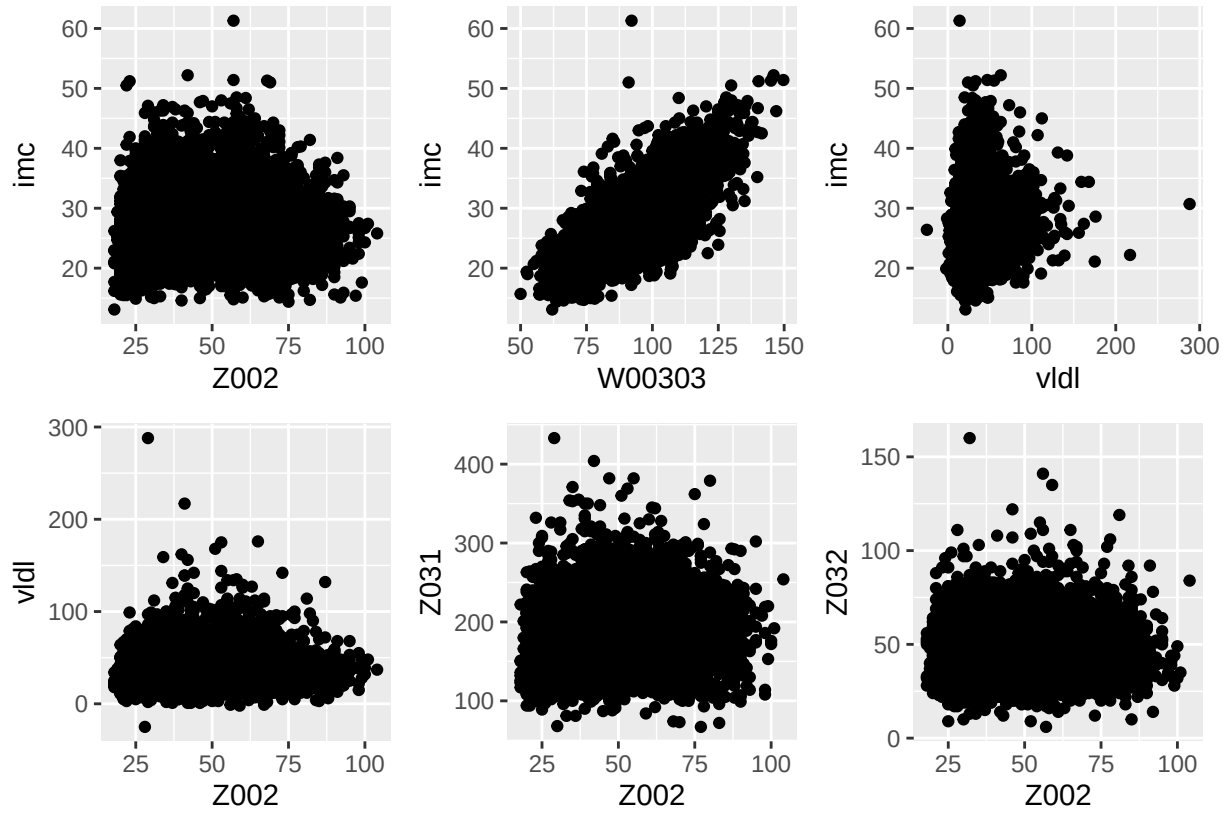
```
library(patchwork)

g1 <- ggplot(banco_trab, aes(diabetes, Z002)) +
  geom_boxplot()

g2 <- ggplot(banco_trab, aes(diabetes, imc)) +
  geom_boxplot()

g3 <- ggplot(banco_trab, aes(diabetes, W00303)) +
  geom_boxplot()

g4 <- ggplot(banco_trab, aes(diabetes, Z031)) +
  geom_boxplot()

g5 <- ggplot(banco_trab, aes(diabetes, vldl)) +
  geom_boxplot()

g6 <- ggplot(banco_trab, aes(diabetes, Z032)) +
  geom_boxplot()

g1 + g2 + g3 + g4 + g5 + g6
```



Gráficos de dispersão

```
W00303 -> Circunferência da cintura parece ter uma correlação linear com o IMC.
```

```
g5 <- ggplot(banco_trab, aes(Z002, imc))+
  geom_point()

g6 <- ggplot(banco_trab, aes(W00303, imc))+
  geom_point()

g7 <- ggplot(banco_trab, aes(vldl, imc))+
  geom_point()

g8 <- ggplot(banco_trab, aes(Z002, vldl))+
  geom_point()

g9 <- ggplot(banco_trab, aes(Z002, Z031))+
  geom_point()

g10 <- ggplot(banco_trab, aes(Z002, Z032))+
  geom_point()

g5 + g6 + g7 + g8 + g9 + g10
```



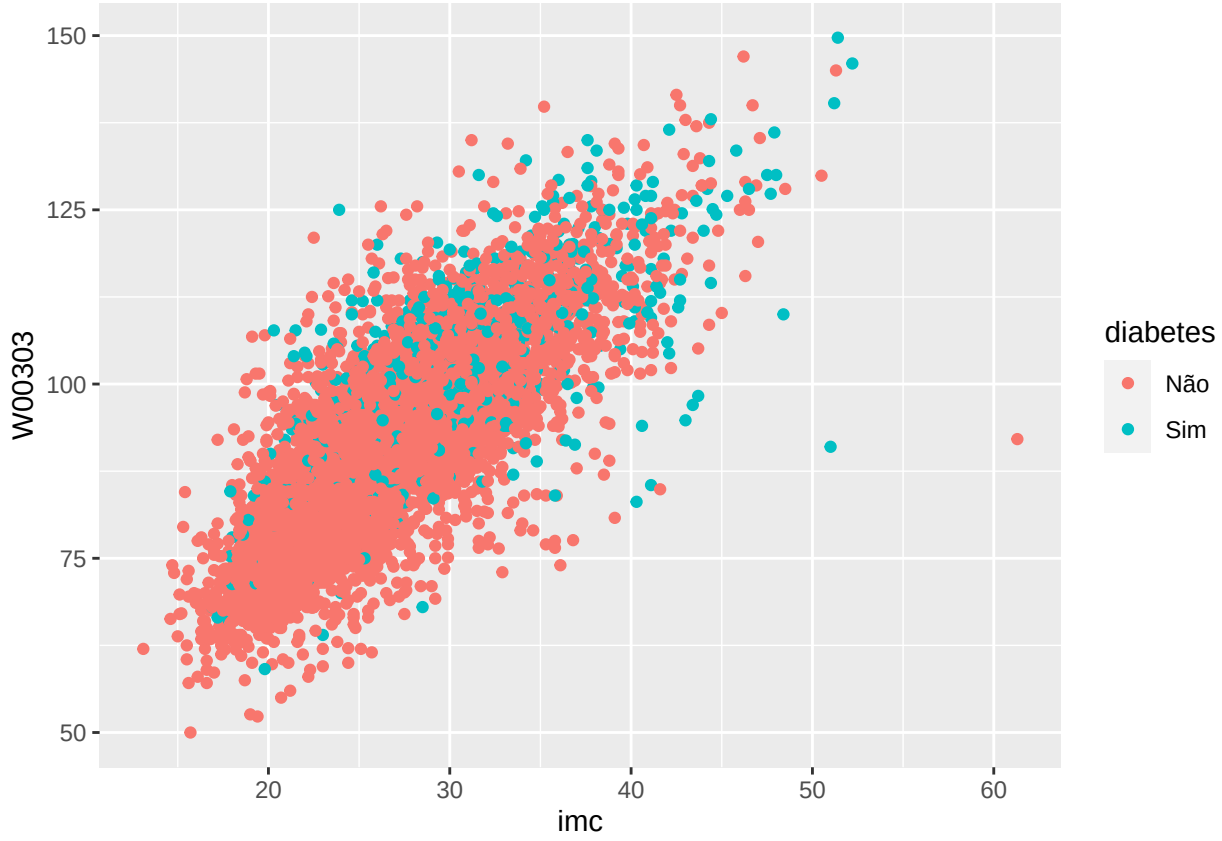
Análise do IMC vs circunferência abdominal por diagnóstico de diabetes

Como esperado, independentemente de diabetes a relação entre

IMC e circunferência abdominal permanece linear. Porém, as pessoas com DM têm valores mais elevados de IMC e circunferência da cintura do que os sem diabetes.

```
banco_trab %>%
  filter(!is.na(diabetes)) %>%
  ggplot(aes(imc, W00303))+
  geom_point(aes(color = diabetes))
```

```
## Warning: Removed 76 rows containing missing values (geom_point).
```

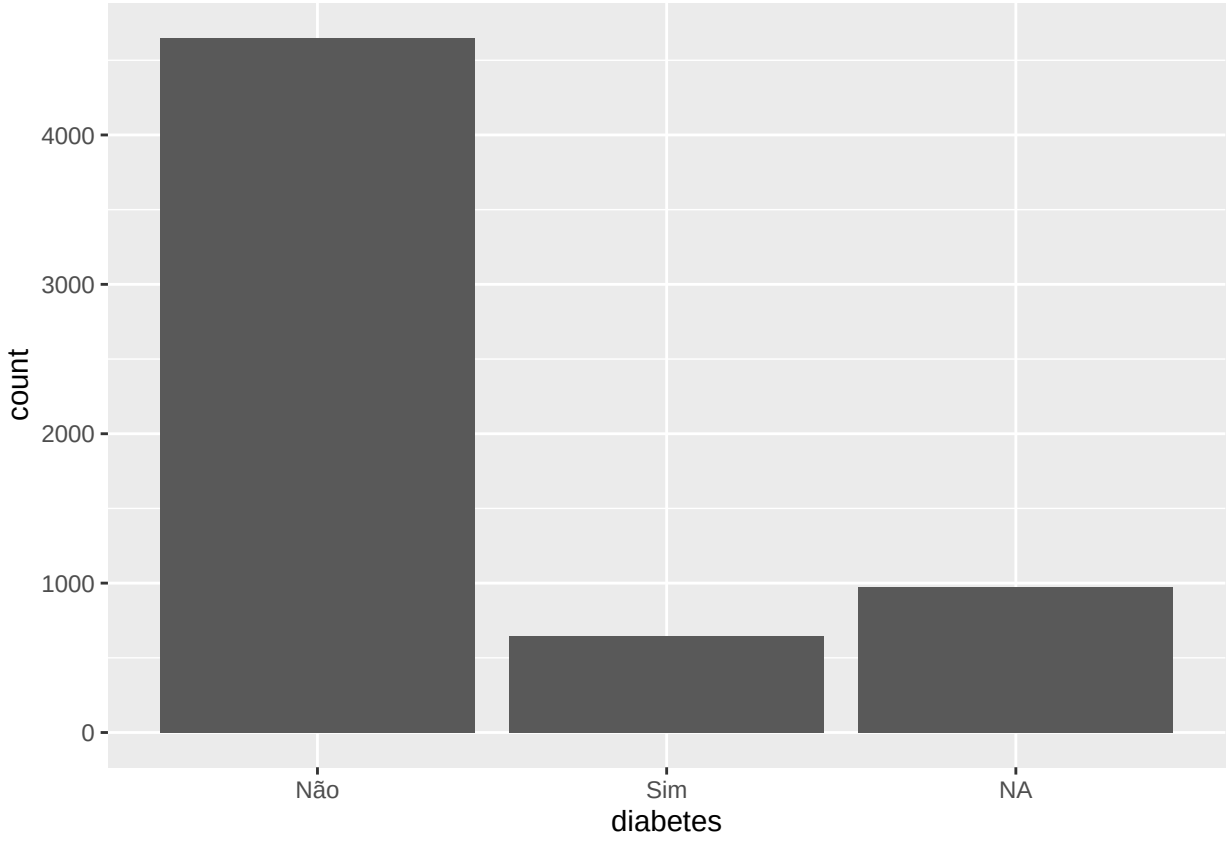


Particionamento dos dados entre treino (70%) e teste(30%)

```
set.seed(789)
particionamento <- caret::createDataPartition(banco_trab$diabetes,
                                              # cria as partições de forma estratificada
                                               p = 0.70,
                                               list = FALSE)
#O parâmetro list = FALSE evita que o resultado seja armazenado em formato de lista.

banco_train <- banco_trab[particionamento, ]
banco_test <- banco_trab[-particionamento, ]
```

Visualização do balanceamento

```
banco_train %>% ggplot(aes(x=diabetes)) + geom_bar(stat="count")
```



Relação diabetes no conjunto treinamento.

```
table(banco_train$diabetes)
```

```
##
##  Não  Sim
## 4651  647
```

Pré-processamento conjunto de treino

Sumário do banco treino.

```
summary(banco_train)
```

```
##    diabetes          Z001          Z002             imc            W00303
##  Length:6269        1:2613   Min.   : 18.00   Min.   :13.10   Min.   : 55.0
##  Class :character   2:3656   1st Qu.: 34.00   1st Qu.:22.95   1st Qu.: 81.5
##  Mode  :character            Median : 45.00   Median :26.00   Median : 90.5
##                             Mean   : 46.97   Mean   :26.58   Mean   : 91.2
```

```
##                              3rd Qu.: 58.00   3rd Qu.:29.40   3rd Qu.: 99.8
##                              Max.   :104.00   Max.   :61.30   Max.   :149.7
##                                               NA's   :58      NA's   :58
##       Z031            Z032             Z033            vldl        VDD004
##  Min.   : 67.0   Min.   :  6.00   Min.   : 22.0   Min.   :-25.00   1:1182
##  1st Qu.:160.0   1st Qu.: 37.00   1st Qu.: 85.0   1st Qu.: 25.00   2:1589
##  Median :184.0   Median : 44.00   Median :104.0   Median : 32.00   3: 671
##  Mean   :187.3   Mean   : 45.88   Mean   :105.7   Mean   : 35.64   4: 319
##  3rd Qu.:212.0   3rd Qu.: 53.00   3rd Qu.:124.0   3rd Qu.: 42.00   5:1600
##  Max.   :404.0   Max.   :160.00   Max.   :261.0   Max.   :217.00   6: 272
##  NA's   :291     NA's   :304      NA's   :292     NA's   :305      7: 636
##   regiao
##  1   :1623
##  2   :2135
##  3   :1065
##  4   : 747
##  5   : 694
##  NA's:   5
##
```

Preenchendo os faltantes pela média

Decidimos filtrar os outliers antes de tirar a média para preencher os dados faltantes.

Obs: IMC > 50 e VLDL > 100.

```
media_imc <- banco_train %>%
  filter(imc < 50)


banco_train$imc[is.na(banco_train$imc)] <- mean(media_imc$imc, na.rm = T)
# preenche pela média sem outliers

media_vldl <- banco_train %>%
  filter(vldl <= 100)

banco_train$vldl[which(is.na(banco_train$vldl))] <- mean(media_vldl$vldl, na.rm = T)
# preenche pela média sem outliers


banco_train$W00303[which(is.na(banco_train$W00303))] <- mean(banco_train$W00303, na.rm = T)

banco_train$Z031[which(is.na(banco_train$Z031))] <- mean(banco_train$Z031, na.rm = T)

banco_train$Z032[is.na(banco_train$Z032)] <- mean(banco_train$Z032, na.rm = T)

banco_train$Z033[is.na(banco_train$Z033)] <- mean(banco_train$Z033, na.rm = T)
```

Observamos 5 dados faltantes na variável "regiao", substituimos pela moda.

```r
banco_train <- banco_train %>%
  mutate(
    regiao = replace_na(regiao, '5')
    # poderia usar essa função (replace_na) para substituir os NAs das outras variáveis.
  )
```

```r
summary(banco_train)
```

```
##    diabetes          Z001          Z002              imc             W00303
##  Length:6269        1:2613   Min.   : 18.00   Min.   :13.10   Min.   : 55.0
##  Class :character   2:3656   1st Qu.: 34.00   1st Qu.:23.00   1st Qu.: 81.5
##  Mode  :character            Median : 45.00   Median :26.10   Median : 90.8
##                             Mean   : 46.97   Mean   :26.58   Mean   : 91.2
##                             3rd Qu.: 58.00   3rd Qu.:29.40   3rd Qu.: 99.6
##                             Max.   :104.00   Max.   :61.30   Max.   :149.7
##
##       Z031            Z032             Z033            vldl         VDD004
##  Min.   : 67.0   Min.   :  6.00   Min.   : 22.0   Min.   :-25.00   1:1182
##  1st Qu.:161.0   1st Qu.: 38.00   1st Qu.: 86.0   1st Qu.: 26.00   2:1589
##  Median :187.0   Median : 45.00   Median :105.0   Median : 33.00   3: 671
##  Mean   :187.3   Mean   : 45.88   Mean   :105.7   Mean   : 35.62   4: 319
##  3rd Qu.:210.0   3rd Qu.: 52.00   3rd Qu.:123.0   3rd Qu.: 41.00   5:1600
##  Max.   :404.0   Max.   :160.00   Max.   :261.0   Max.   :217.00   6: 272
##                                                                    7: 636
##  regiao
##  1:1623
##  2:2135
##  3:1065
##  4: 747
##  5: 699
##
##
```

Retirando as observações com 'missing' na variável desfecho

Retirar dos dois bancos.

```r
banco_train <- banco_train %>%
  filter(!is.na(diabetes))
```

Padronização das variáveis numéricas

Preferimos criar um banco com as variáveis numéricas padronizadas,
testando e comparando com o não padronizado, que é mais fácil de interpretar.

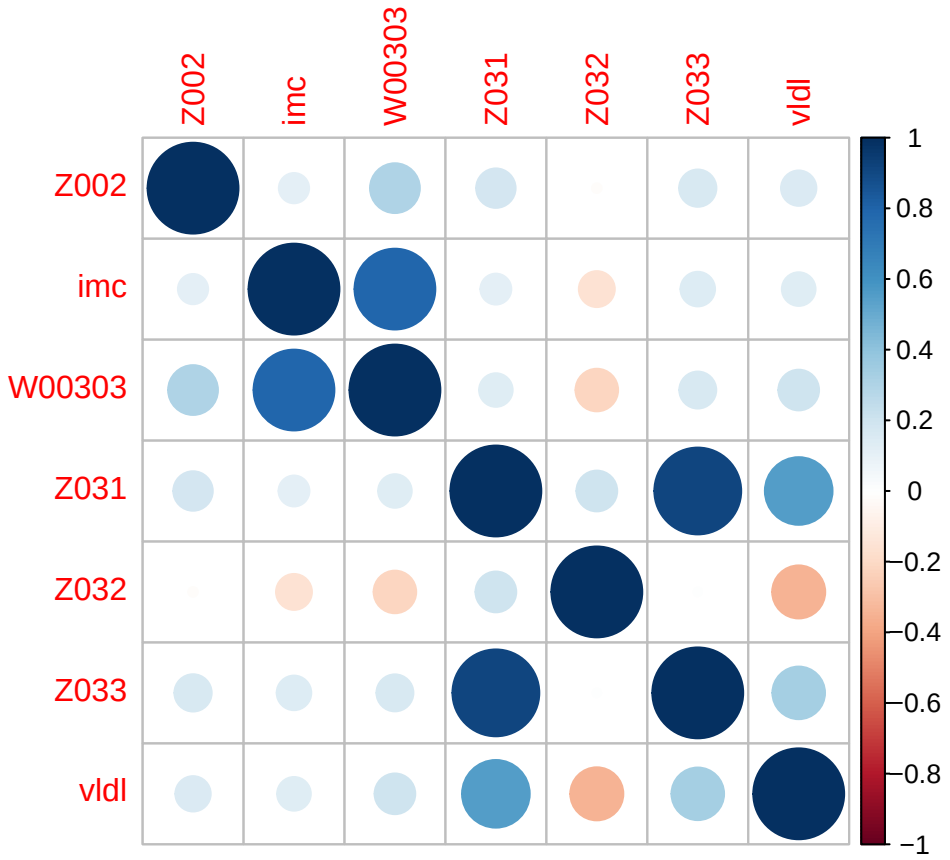
```r
banco_train_pad <- banco_train %>%
  mutate(
    Z002 = scale(Z002),
    imc = scale(imc),
    W00303 = scale(W00303),
```

```
    Z031 = scale(Z031),
    Z032 = scale(Z032),
    Z033 = scale(Z033),
    vldl = scale(vldl)
    )
```

Correlação entre variáveis numéricas padronizadas.

O colesterol total (Z031) tem uma alta relação com o LDL (Z033), como era de se esperar.
A correlação negativa entre vldl e HDL e LDL era esperada também, já que aquela
é resultado da subtração do colesterol total pelas duas últimas.
A cirurncferência abdominal e o IMC também apresentaram uma correlação bem elevada.

```
correl <- banco_train_pad %>%
  select(where(is.numeric)) %>%
  cor(use = "pairwise")  # não estava funcionando da outra forma (sem o 'pairwise').

corrplot::corrplot(correl, method = 'circle')
```



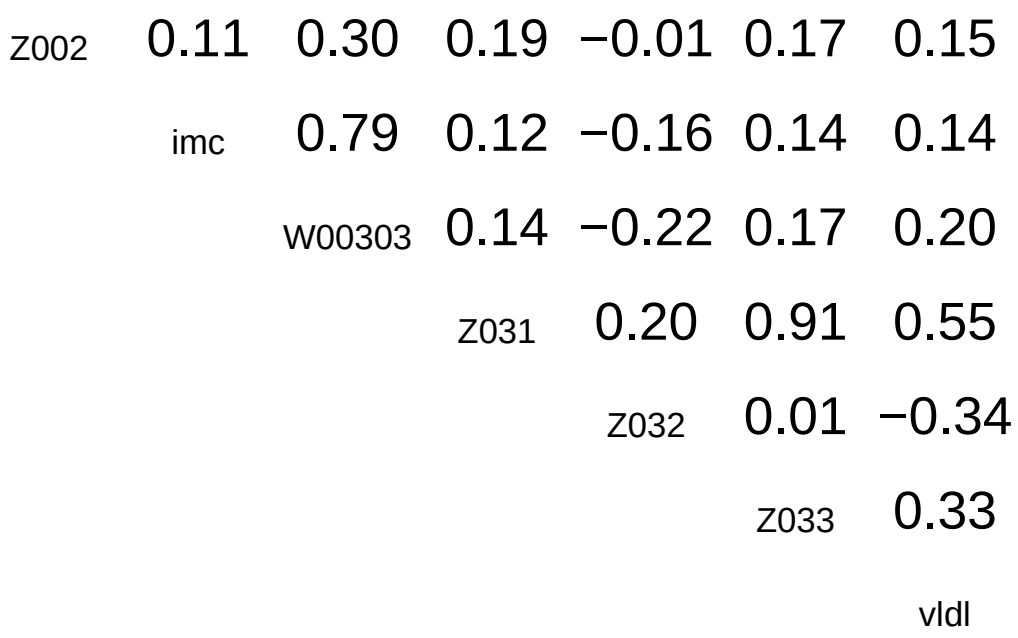
A correção entre Colesterol total e LDL é de mais de 90% vamos analisar qual deve
ser retirada do modelo. Talvez o melhor seja retirar o colesterol total, pois é
resultado da soma de três variáveis (VLD, HDL E LDL).
A correlação entre IMC e circubferência abdominal é de 79%, vamos testar as duas no
modelo, mas é possível que precise retirar uma delas.

```
library(corrgram)
```

```
##
## Attaching package: 'corrgram'
```

```
## The following object is masked from 'package:lattice':
##
##     panel.fill
```

```
corrgram::corrgram(correl, lower.panel = panel.pts, upper.panel= panel.conf,
                   diag.panel = panel.density)
```



Associação entre variáveis numéricas e desfecho

A única variável sem significância estatística foi o LDL.

```
t.test(banco_train_pad$Z002 ~ banco_train_pad$diabetes) #idade
```

```
##
##  Welch Two Sample t-test
##
## data:  banco_train_pad$Z002 by banco_train_pad$diabetes
```

```
## t = -18.68, df = 872.55, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group Não and group Sim is not equal to 0
## 95 percent confidence interval:
##  -0.7907109 -0.6403535
## sample estimates:
## mean in group Não mean in group Sim
##        -0.0873819         0.6281503
```

```
t.test(banco_train_pad$imc ~ banco_train_pad$diabetes)
```

```
##
##  Welch Two Sample t-test
##
## data:  banco_train_pad$imc by banco_train_pad$diabetes
## t = -11.999, df = 776.61, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group Não and group Sim is not equal to 0
## 95 percent confidence interval:
##  -0.6589675 -0.4736696
## sample estimates:
## mean in group Não mean in group Sim
##        -0.0691597         0.4971588
```

```
t.test(banco_train_pad$W00303 ~ banco_train_pad$diabetes) #circunferência cintura
```

```
##
##  Welch Two Sample t-test
##
## data:  banco_train_pad$W00303 by banco_train_pad$diabetes
## t = -16.872, df = 820.41, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group Não and group Sim is not equal to 0
## 95 percent confidence interval:
##  -0.7927467 -0.6275139
## sample estimates:
## mean in group Não mean in group Sim
##       -0.08672221        0.62340805
```

```
t.test(banco_train_pad$Z031 ~ banco_train_pad$diabetes) # Colesterol total
```

```
##
##  Welch Two Sample t-test
##
## data:  banco_train_pad$Z031 by banco_train_pad$diabetes
## t = -2.2792, df = 801.35, p-value = 0.02292
## alternative hypothesis: true difference in means between group Não and group Sim is not equal to 0
## 95 percent confidence interval:
##  -0.19129776 -0.01426037
## sample estimates:
## mean in group Não mean in group Sim
##       -0.01255154        0.09022752
```

```r
t.test(banco_train_pad$Z032 ~ banco_train_pad$diabetes) # HDL
```

```
##
##  Welch Two Sample t-test
##
## data:  banco_train_pad$Z032 by banco_train_pad$diabetes
## t = 6.3156, df = 834.08, p-value = 4.373e-10
## alternative hypothesis: true difference in means between group Não and group Sim is not equal to 0
## 95 percent confidence interval:
##  0.1826274 0.3473332
## sample estimates:
## mean in group Não mean in group Sim
##        0.03235981       -0.23262052
```

```r
t.test(banco_train_pad$Z033 ~ banco_train_pad$diabetes) # LDL
```

```
##
##  Welch Two Sample t-test
##
## data:  banco_train_pad$Z033 by banco_train_pad$diabetes
## t = -1.6777, df = 803.84, p-value = 0.0938
## alternative hypothesis: true difference in means between group Não and group Sim is not equal to 0
## 95 percent confidence interval:
##  -0.16327847  0.01279246
## sample estimates:
## mean in group Não mean in group Sim
##      -0.009188793       0.066054212
```

```r
t.test(banco_train_pad$vldl ~ banco_train_pad$diabetes)
```

```
##
##  Welch Two Sample t-test
##
## data:  banco_train_pad$vldl by banco_train_pad$diabetes
## t = -6.5891, df = 767.57, p-value = 8.22e-11
## alternative hypothesis: true difference in means between group Não and group Sim is not equal to 0
## 95 percent confidence interval:
##  -0.4191019 -0.2266999
## sample estimates:
## mean in group Não mean in group Sim
##       -0.03943316        0.28346774
```

Teste de variáveis categóricas

Todas as variáveis categóricas tiveram significância estatística.

```r
chisq.test(banco_train_pad$Z001, banco_train_pad$diabetes) #sexo
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
```

```
##
## data:  banco_train_pad$Z001 and banco_train_pad$diabetes
## X-squared = 8.8551, df = 1, p-value = 0.002923
```

```
chisq.test(banco_train_pad$VDD004, banco_train_pad$diabetes) # grau de instrução
```

```
##
##  Pearson's Chi-squared test
##
## data:  banco_train_pad$VDD004 and banco_train_pad$diabetes
## X-squared = 85.596, df = 6, p-value = 2.484e-16
```

```
chisq.test(banco_train_pad$regiao, banco_train_pad$diabetes) # região do Brasil.
```

```
##
##  Pearson's Chi-squared test
##
## data:  banco_train_pad$regiao and banco_train_pad$diabetes
## X-squared = 15.869, df = 4, p-value = 0.0032
```

```
chisq.test(banco_train_pad$regiao, banco_train_pad$VDD004)
```

```
##
##  Pearson's Chi-squared test
##
## data:  banco_train_pad$regiao and banco_train_pad$VDD004
## X-squared = 179.81, df = 24, p-value < 2.2e-16
```

Tratamento do desbalanceamento na variável desfecho

```
table(banco_train_pad$diabetes)
```

```
##
##  Não  Sim
## 4651  647
```

Utilização do 'SMOTE'do pacote 'smotefamily' para criar observações

Obs: tem que usar somente variáveis numéricas.

```
banco_train_pad <- banco_train_pad %>%
  mutate(
    diabetes = if_else(diabetes == "Sim", 1, 2),
    Z001 = as.numeric(Z001),
    VDD004 = as.numeric(VDD004),
    regiao = as.numeric(regiao)
  )
```

Criando observações para equilibrar o desfecho no banco.

```
banco_smote <- smotefamily::SMOTE(
  banco_train_pad[, 2:11],
  unlist(as.numeric(banco_train_pad$diabetes)),
  K = 5, dup_size = 6
)
```

```
banco_smote <- banco_smote$data
```

Desfecho mais equilibrado.

```
table(banco_smote$class) # a variável passa a se chamar 'class'
```

```
##
##    1    2
## 4529 4651
```

Retorna para o nome original e fatores.

```
banco_train_pad <- banco_smote %>%
  mutate(
    Z001 = if_else(Z001 == 1, "M", "F"),
    class = if_else(class == 1, "Sim", "Não"),
    VDD004 = as.factor(round(VDD004)),
    regiao = as.factor(round(regiao))
  )
```

Modelagem para regressão logística

- **Modelo 1** - sem LDL e circunferência abdominal.
- **Modelo 2** - sem LDL e IMC.

  O modelo com a circunferência abdominal parece um pouco melhor do que o com IMC.

- **Modelo 3** - sem Colesterol, LDL e IMC.
- **MOdelo 4** - com colesterol. Sem LDL, HDL e IMC.

```
mod_1 <- train(class ~ Z001 + Z002 + imc + Z031 +Z032  + VDD004 + regiao +
                       vldl ,
                     data = banco_train_pad,
                     method = "glm",
                     family = binomial(link = "logit")
)

  mod_2 <- train(class ~ Z001 + Z002 + W00303 + Z031 +Z032  + VDD004 + # HDL e Colesterol
                   regiao + vldl , # com circunferência abdominal
                     data = banco_train_pad,
                     method = "glm",
```

```
                      family = binomial(link = "logit")
)

  mod_3 <- train(class ~ Z001 + Z002 + W00303  + Z032  + VDD004 + regiao + # com HDL
                       vldl ,
                     data = banco_train_pad,
                     method = "glm",
                     family = binomial(link = "logit")
)

  mod_4 <- train(class ~ Z001 + Z002 + W00303  + Z031  + VDD004 + regiao + # com colesterol
                   vldl ,
                 data = banco_train_pad,
                 method = "glm",
                 family = binomial(link = "logit")
)

  mod_5 <- train(class ~ Z002 + W00303  + Z031  + VDD004 + regiao + # Sem o gênero
                   vldl ,
                 data = banco_train_pad,
                 method = "glm",
                 family = binomial(link = "logit")
)
```

- Sumário de modelos

Modelo 1

```
summary(mod_1)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7140  -0.9238  -0.3142   0.9334   2.4963
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.28233    0.07270  -3.884 0.000103 ***
## Z001M       -0.92295    0.05386 -17.137  < 2e-16 ***
## Z002         0.79967    0.02996  26.694  < 2e-16 ***
## imc          0.47666    0.02531  18.831  < 2e-16 ***
## Z031        -0.16200    0.03423  -4.733 2.21e-06 ***
## Z032        -0.23652    0.03360  -7.040 1.93e-12 ***
## VDD0042      0.02344    0.07003   0.335 0.737879
## VDD0043      0.14977    0.09214   1.625 0.104065
## VDD0044     -0.27026    0.13953  -1.937 0.052758 .
## VDD0045     -0.15523    0.07732  -2.008 0.044689 *
## VDD0046     -0.25151    0.15115  -1.664 0.096115 .
## VDD0047     -0.12648    0.09580  -1.320 0.186753
```

```
## regiao2      0.25590    0.06779   3.775 0.000160 ***
## regiao3      0.26291    0.07803   3.369 0.000753 ***
## regiao4      0.22294    0.08534   2.612 0.008996 **
## regiao5      0.32428    0.08792   3.689 0.000226 ***
## vldl         0.23838    0.03531   6.751 1.46e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12725  on 9179  degrees of freedom
## Residual deviance: 10329  on 9163  degrees of freedom
## AIC: 10363
##
## Number of Fisher Scoring iterations: 4
```

Modelo 2

```
summary(mod_2)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7607  -0.9068  -0.2895   0.9049   2.5874
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.17951    0.07340  -2.446  0.01446 *
## Z001M       -1.14546    0.05506 -20.804  < 2e-16 ***
## Z002         0.66624    0.03053  21.823  < 2e-16 ***
## W00303       0.64200    0.02813  22.826  < 2e-16 ***
## Z031        -0.17045    0.03448  -4.943 7.68e-07 ***
## Z032        -0.19208    0.03412  -5.629 1.81e-08 ***
## VDD0042      0.02715    0.07086   0.383  0.70159
## VDD0043      0.14204    0.09319   1.524  0.12745
## VDD0044     -0.31495    0.14159  -2.224  0.02612 *
## VDD0045     -0.14774    0.07810  -1.892  0.05854 .
## VDD0046     -0.27210    0.15387  -1.768  0.07701 .
## VDD0047     -0.10175    0.09697  -1.049  0.29406
## regiao2      0.18014    0.06846   2.631  0.00850 **
## regiao3      0.18855    0.07883   2.392  0.01676 *
## regiao4      0.09839    0.08663   1.136  0.25607
## regiao5      0.26124    0.08883   2.941  0.00327 **
## vldl         0.22316    0.03552   6.283 3.32e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12725  on 9179  degrees of freedom
```

```
## Residual deviance: 10142  on 9163  degrees of freedom
## AIC: 10176
##
## Number of Fisher Scoring iterations: 4
```

Modelo 3

```
summary(mod_3)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7216  -0.9090  -0.2873   0.8961   2.6346
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.17290    0.07330  -2.359  0.01834 *
## Z001M       -1.12241    0.05468 -20.526  < 2e-16 ***
## Z002         0.66344    0.03046  21.777  < 2e-16 ***
## W00303       0.62889    0.02793  22.518  < 2e-16 ***
## Z032        -0.27623    0.02971  -9.298  < 2e-16 ***
## VDD0042      0.01961    0.07072   0.277  0.78157
## VDD0043      0.13838    0.09328   1.483  0.13795
## VDD0044     -0.31409    0.14153  -2.219  0.02647 *
## VDD0045     -0.14472    0.07795  -1.857  0.06336 .
## VDD0046     -0.28014    0.15377  -1.822  0.06848 .
## VDD0047     -0.11112    0.09681  -1.148  0.25105
## regiao2      0.16447    0.06824   2.410  0.01595 *
## regiao3      0.16719    0.07856   2.128  0.03333 *
## regiao4      0.08625    0.08645   0.998  0.31843
## regiao5      0.25361    0.08868   2.860  0.00424 **
## vldl         0.10405    0.02567   4.053 5.06e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12725  on 9179  degrees of freedom
## Residual deviance: 10166  on 9164  degrees of freedom
## AIC: 10198
##
## Number of Fisher Scoring iterations: 4
```

Modelo 4

```
summary(mod_4)
```

```
##
## Call:
```

```
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7896  -0.9072  -0.2963   0.9162   2.5743
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.17054    0.07325  -2.328  0.01990 *
## Z001M       -1.11079    0.05455 -20.363  < 2e-16 ***
## Z002         0.65848    0.03043  21.637  < 2e-16 ***
## W00303       0.67314    0.02764  24.355  < 2e-16 ***
## Z031        -0.26841    0.02992  -8.970  < 2e-16 ***
## VDD0042      0.02284    0.07066   0.323  0.74646
## VDD0043      0.13386    0.09287   1.441  0.14950
## VDD0044     -0.30440    0.14115  -2.157  0.03104 *
## VDD0045     -0.16579    0.07788  -2.129  0.03328 *
## VDD0046     -0.28578    0.15355  -1.861  0.06272 .
## VDD0047     -0.12172    0.09663  -1.260  0.20780
## regiao2      0.17895    0.06840   2.616  0.00889 **
## regiao3      0.19153    0.07874   2.432  0.01500 *
## regiao4      0.08142    0.08639   0.942  0.34595
## regiao5      0.26315    0.08860   2.970  0.00298 **
## vldl         0.33559    0.02992  11.218  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12725  on 9179  degrees of freedom
## Residual deviance: 10174  on 9164  degrees of freedom
## AIC: 10206
##
## Number of Fisher Scoring iterations: 4
```

Modelo 5

```
summary(mod_5)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5517  -0.9545  -0.3717   0.9586   2.2645
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.43705    0.07041  -6.207 5.41e-10 ***
## Z002         0.65159    0.02958  22.026  < 2e-16 ***
## W00303       0.60274    0.02658  22.674  < 2e-16 ***
## Z031        -0.16697    0.02854  -5.850 4.92e-09 ***
```

```
## VDD0042     -0.06686    0.06819  -0.980 0.326844
## VDD0043      0.07863    0.09023   0.871 0.383536
## VDD0044     -0.46775    0.13894  -3.367 0.000761 ***
## VDD0045     -0.21711    0.07572  -2.867 0.004141 **
## VDD0046     -0.44139    0.15179  -2.908 0.003640 **
## VDD0047     -0.23973    0.09475  -2.530 0.011406 *
## regiao2      0.17591    0.06680   2.633 0.008457 **
## regiao3      0.19456    0.07681   2.533 0.011307 *
## regiao4      0.10653    0.08461   1.259 0.207999
## regiao5      0.27844    0.08644   3.221 0.001277 **
## vldl         0.24512    0.02842   8.624  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12725  on 9179  degrees of freedom
## Residual deviance: 10614  on 9165  degrees of freedom
## AIC: 10644
##
## Number of Fisher Scoring iterations: 3
```

Predição no conjunto de treino

Identificamos que os melhores modelos são o 2 e o 4.
Como a diferença entre o AIC é pequena e o 4 tem uma variável a menos, optamos por utilizar o 4.
Retiramos as variáveis que não fazem parte do modelo.

```
banco_train_pad$Z032 <- NULL

banco_train_pad$Z033 <- NULL

banco_train_pad$Z032 <- NULL

banco_train_pad$imc <- NULL

yp_treino <- predict(mod_4, newdata = banco_train_pad[, 1:8])

table(banco_train_pad$class, yp_treino)
```

```
##      yp_treino
##        Não  Sim
##   Não 3313 1338
##   Sim 1252 3277
```

Matriz de confusão no treino

Aplicando o modelo nos dados de treinamento, encontramos uma sensibilidade de 71%, especificidade de 72% e acurácia de 72%.

```
caret::confusionMatrix(as.factor(banco_train_pad$class), yp_treino,
                       positive = "Sim")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  Não  Sim
##        Não 3313 1338
##        Sim 1252 3277
##
##                Accuracy : 0.7179
##                  95% CI : (0.7085, 0.7271)
##     No Information Rate : 0.5027
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.4358
##
##  Mcnemar's Test P-Value : 0.09488
##
##             Sensitivity : 0.7101
##             Specificity : 0.7257
##          Pos Pred Value : 0.7236
##          Neg Pred Value : 0.7123
##              Prevalence : 0.5027
##          Detection Rate : 0.3570
##    Detection Prevalence : 0.4934
##       Balanced Accuracy : 0.7179
##
##        'Positive' Class : Sim
##
```

Pré-processamento conjunto de teste

Manipulando os Nas com os mesmo critérios do treino, porém com os dados do teste

```
# media_imc <- banco_test %>%
#   filter(imc < 50)
#
# banco_test$imc[is.na(banco_test$imc)] <- mean(media_imc$imc, na.rm = T)
# # preenche pela média sem outliers
#

media_vldl <- banco_test %>%   ## O IMC não faz parte do modelo escolhido.
  filter(vldl <= 100)

banco_test$vldl[which(is.na(banco_test$vldl))] <- mean(media_vldl$vldl, na.rm = T)
# preenche pela média sem outliers


banco_test$W00303[which(is.na(banco_test$W00303))] <- mean(banco_test$W00303, na.rm = T)
```

```
banco_test$Z031[which(is.na(banco_test$Z031))] <- mean(banco_test$Z031, na.rm = T)

# banco_test$Z032[is.na(banco_test$Z032)] <- mean(banco_test$Z032, na.rm = T)
#
# banco_test$Z033[is.na(banco_test$Z033)] <- mean(banco_test$Z033, na.rm = T)

## LDL e HDL não fazem parte do modelo escolhido


banco_test <- banco_test %>%
  select(-imc, -Z032, -Z033) %>%
  mutate(
    regiao = replace_na(regiao, '5')
    # poderia usar essa função (replace_na) para substituir os NAs das outras variáveis.
  )
```

```
summary(banco_test)
```

```
##    diabetes          Z001          Z002            W00303            Z031
##  Length:2683        1:1112   Min.   : 19.00   Min.   : 50.00   Min.   : 72.0
##  Class :character   2:1571   1st Qu.: 33.50   1st Qu.: 81.85   1st Qu.:161.0
##  Mode  :character            Median : 45.00   Median : 90.50   Median :186.0
##                             Mean   : 46.55   Mean   : 90.92   Mean   :186.3
##                             3rd Qu.: 58.00   3rd Qu.: 99.85   3rd Qu.:208.0
##                             Max.   :101.00   Max.   :140.00   Max.   :433.0
##
##       vldl        VDD004  regiao
##  Min.   : -2.00   1:503   1:674
##  1st Qu.: 25.00   2:704   2:918
##  Median : 33.00   3:270   3:449
##  Mean   : 35.23   4:124   4:328
##  3rd Qu.: 40.00   5:691   5:314
##  Max.   :288.00   6:108
##                   7:283
```

Retirando as observações NA do desfecho no teste.

```
banco_test <- banco_test %>%
  filter(!is.na(diabetes))
```

Padronização das variáveis numéricas no banco de teste.

```
banco_test_pad <- banco_test %>%
  mutate(
    Z001 = if_else(Z001 == 1, "M", "F"),
    Z002 = scale(Z002),
    # imc = scale(imc),
    W00303 = scale(W00303),
    Z031 = scale(Z031),
    # Z032 = scale(Z032),
    # Z033 = scale(Z033),
    vldl = scale(vldl)
    )
```

Predição e matriz de confusão no conjunto de TESTE

```
yp <- predict(mod_4, newdata = banco_test_pad[, 1:8])

table(banco_test_pad$diabetes, yp)
```

```
##      yp
##        Não  Sim
##   Não 1427  565
##   Sim   88  188
```

Matriz de confusão dos dados de teste.

No nosso banco de teste, a acurácia manteve-se estável, de 72% para 71%, em relação ao banco de treinamento. Porém, houve uma redução da sensibilidade de 71% para 67%, mas estabilidade da especificidade em 72%.
Talvez tenha ocorrido um overfitting, mas os resultados são muito semelhantes.
Parece um bom modelo para identificar pessoas com alta probabilidade de ter diabetes.

```
caret::confusionMatrix(yp, as.factor(banco_test_pad$diabetes) , positive = "Sim")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  Não  Sim
##        Não 1427   88
##        Sim  565  188
##
##                Accuracy : 0.7121
##                  95% CI : (0.693, 0.7307)
##     No Information Rate : 0.8783
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2279
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.68116
##             Specificity : 0.71637
##          Pos Pred Value : 0.24967
##          Neg Pred Value : 0.94191
##              Prevalence : 0.12169
##          Detection Rate : 0.08289
##    Detection Prevalence : 0.33201
##       Balanced Accuracy : 0.69876
##
##        'Positive' Class : Sim
##
```

Árvore de decisão

Vamos utilizar todas as variáveis incialmente.

Pré-processamento dos dados

Optamos utilizar os dados sem padronizar para melhorar a interpretação.

Pré-processamento treino

```
banco_train_smote <- banco_train %>%
  mutate(
    diabetes = if_else(diabetes == "Sim", 1, 2),
    Z001 = as.numeric(Z001),
    VDD004 = as.numeric(VDD004),
    regiao = as.numeric(regiao)
  )


smote_train <- smotefamily::SMOTE(
  banco_train_smote[, 2:11],
  unlist(as.numeric(banco_train_smote$diabetes)),
  K = 5, dup_size = 6
)


smote_train <- smote_train$data

table(smote_train$class)
```

```
##
##    1    2
## 4529 4651
```

```
banco_train_arv <- smote_train %>%
  mutate(
    Z001 = as.factor(if_else(Z001 == 1, "M", "F") ),
    class = as.factor(if_else(class == 1, "Sim", "Não") ),
    VDD004 = as.factor(round(VDD004)),
    regiao = as.factor(round(regiao))
  ) %>%
  select(-imc, -Z032, -Z033)

summary(banco_train_arv)
```

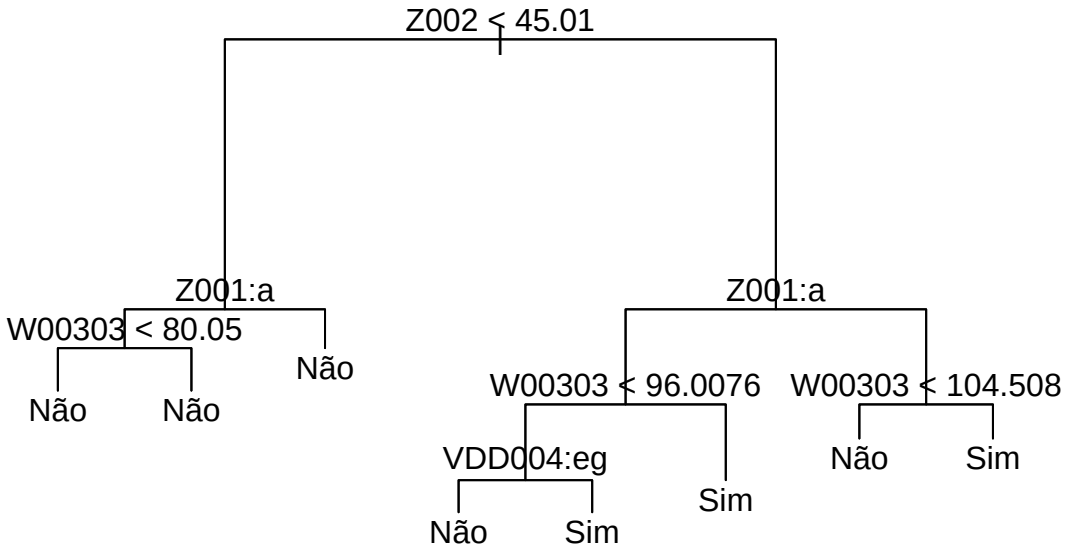
```
##  Z001          Z002            W00303           Z031            vldl
##  F:6593   Min.   : 18.00   Min.   : 55.00   Min.   : 67.0   Min.   : -1.00
##  M:2587   1st Qu.: 40.00   1st Qu.: 86.00   1st Qu.:163.0   1st Qu.: 27.00
##           Median : 53.00   Median : 95.09   Median :187.3   Median : 34.02
##           Mean   : 52.51   Mean   : 95.33   Mean   :189.2   Mean   : 37.34
##           3rd Qu.: 64.34   3rd Qu.:104.20   3rd Qu.:213.6   3rd Qu.: 44.00
##           Max.   :104.00   Max.   :149.70   Max.   :404.0   Max.   :217.00
##
##  VDD004   regiao   class
##  1:1767   1:1787   Não:4651
```

```
##  2:2538   2:3093   Sim:4529
##  3:1219   3:2030
##  4: 728   4:1328
##  5:1828   5: 942
##  6: 403
##  7: 697
```

```
library(tree)

Arvore_dec_mod_4 <- tree::tree(class~.,data = banco_train_arv, method = 'class')

summary(Arvore_dec_mod_4)
```

```
##
## Classification tree:
## tree::tree(formula = class ~ ., data = banco_train_arv, method = "class")
## Variables actually used in tree construction:
## [1] "Z002"   "Z001"   "W00303" "VDD004"
## Number of terminal nodes:  8
## Residual mean deviance:  1.073 = 9841 / 9172
## Misclassification error rate: 0.2573 = 2362 / 9180
```

```
plot(Arvore_dec_mod_4)

text(Arvore_dec_mod_4)
```



```
Arvore_dec_mod_4
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 9180 12720.0 Não ( 0.50664 0.49336 )
##    2) Z002 < 45.01 3148  3438.0 Não ( 0.76429 0.23571 )
##      4) Z001: F 2142  2628.0 Não ( 0.69701 0.30299 )
##        8) W00303 < 80.05 552   307.0 Não ( 0.92029 0.07971 ) *
##        9) W00303 > 80.05 1590  2113.0 Não ( 0.61950 0.38050 ) *
##      5) Z001: M 1006   620.0 Não ( 0.90755 0.09245 ) *
##    3) Z002 > 45.01 6032  7964.0 Sim ( 0.37218 0.62782 )
##      6) Z001: F 4451  5370.0 Sim ( 0.29117 0.70883 )
##       12) W00303 < 96.0076 1883  2587.0 Sim ( 0.44450 0.55550 )
##         24) VDD004: 5,7 406   484.0 Não ( 0.71675 0.28325 ) *
##         25) VDD004: 1,2,3,4,6 1477  1946.0 Sim ( 0.36967 0.63033 ) *
##       13) W00303 > 96.0076 2568  2411.0 Sim ( 0.17874 0.82126 ) *
##      7) Z001: M 1581  2128.0 Não ( 0.60025 0.39975 )
##       14) W00303 < 104.508 1053  1262.0 Não ( 0.71320 0.28680 ) *
##       15) W00303 > 104.508 528   698.6 Sim ( 0.37500 0.62500 ) *
```

Existe um desbalanceamento entre os sexos, com mais mulheres com critérios de diabetes.

```
table(banco_train_arv$class, banco_train_arv$Z001 )
```

```
##
##          F    M
##   Não 2789 1862
##   Sim 3804  725
```

Árvore banco teste

```
banco_test_arv <- banco_test %>%
  mutate(
    Z001 = as.factor(if_else(Z001 == 1, "M", "F") ),
    diabetes = as.factor(diabetes),
    VDD004 = as.factor(VDD004),
    regiao = as.factor(regiao)
  )
```

Tabela previsão vs teste.

```
Arv_dec_test_pred <- predict(Arvore_dec_mod_4, banco_test_arv[, 1:8], type = 'class')

table(banco_test_arv$diabetes, Arv_dec_test_pred)
```

```
##      Arv_dec_test_pred
##        Não  Sim
##   Não 1461  531
##   Sim  110  166
```

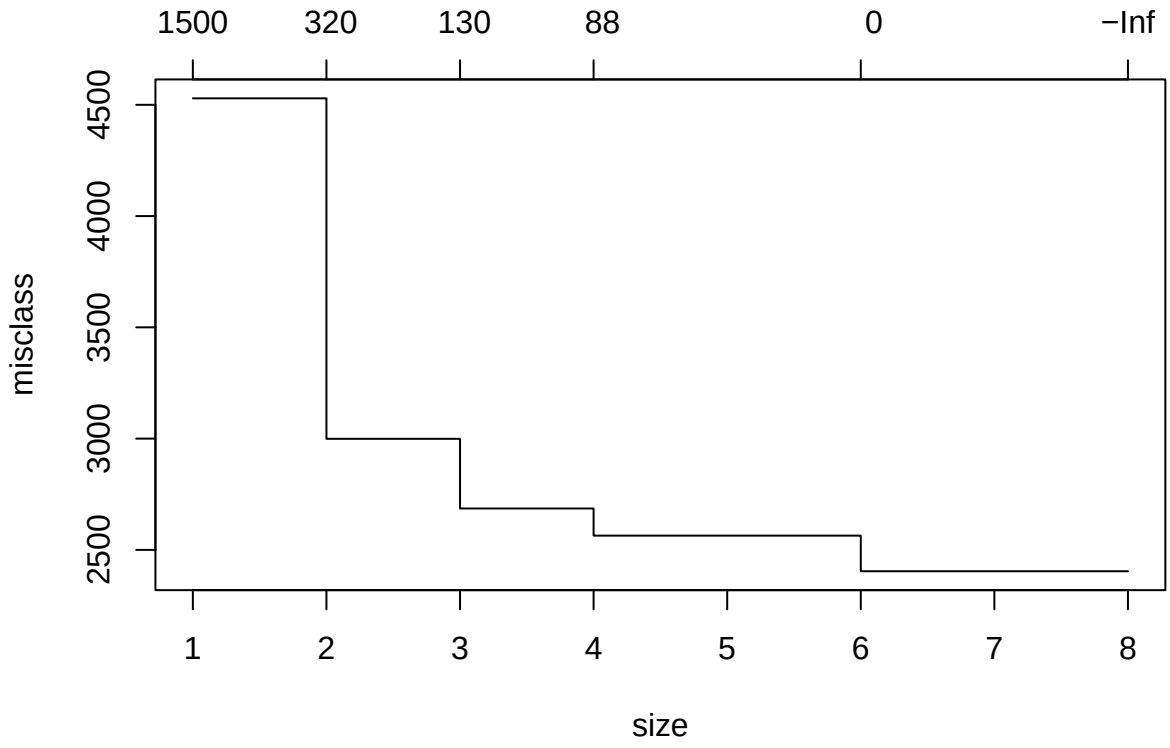
Matriz de confusão no teste.

```
caret::confusionMatrix(Arv_dec_test_pred, banco_test_arv$diabetes, positive = "Sim")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  Não  Sim
##        Não 1461  110
##        Sim  531  166
##
##                Accuracy : 0.7174
##                  95% CI : (0.6983, 0.7358)
##     No Information Rate : 0.8783
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2021
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.60145
##             Specificity : 0.73343
##          Pos Pred Value : 0.23816
##          Neg Pred Value : 0.92998
##              Prevalence : 0.12169
##          Detection Rate : 0.07319
##    Detection Prevalence : 0.30732
##       Balanced Accuracy : 0.66744
##
##        'Positive' Class : Sim
##
```

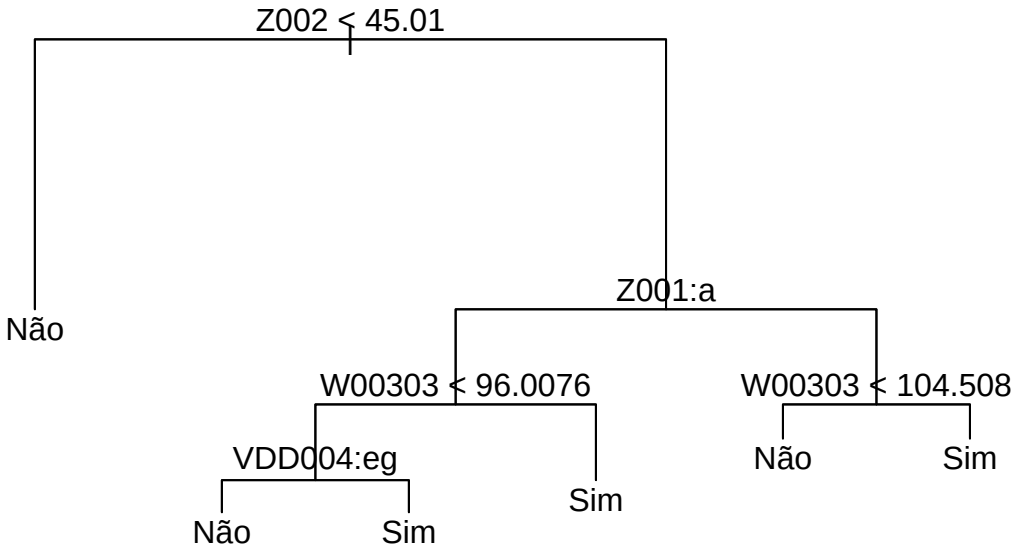
Arvore de decisão do teste.

```
Cross_validation <- tree::cv.tree(Arvore_dec_mod_4, FUN=prune.misclass, K = 10)

plot(Cross_validation)
```



Árvore podada.

```
ArvorePodada = prune.misclass(Arvore_dec_mod_4, best =6)

plot(ArvorePodada)
text(ArvorePodada)
```




```
ArvorePodada
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 9180 12720.0 Não ( 0.5066 0.4934 )
##    2) Z002 < 45.01 3148  3438.0 Não ( 0.7643 0.2357 ) *
##    3) Z002 > 45.01 6032  7964.0 Sim ( 0.3722 0.6278 )
##      6) Z001: F 4451  5370.0 Sim ( 0.2912 0.7088 )
##       12) W00303 < 96.0076 1883  2587.0 Sim ( 0.4445 0.5555 )
##         24) VDD004: 5,7 406   484.0 Não ( 0.7167 0.2833 ) *
##         25) VDD004: 1,2,3,4,6 1477  1946.0 Sim ( 0.3697 0.6303 ) *
##       13) W00303 > 96.0076 2568  2411.0 Sim ( 0.1787 0.8213 ) *
##      7) Z001: M 1581  2128.0 Não ( 0.6003 0.3997 )
##       14) W00303 < 104.508 1053  1262.0 Não ( 0.7132 0.2868 ) *
##       15) W00303 > 104.508 528   698.6 Sim ( 0.3750 0.6250 ) *
```

Considerações finais

- O R não trabalha muito bem com "*character*"; é melhor trabalhar com as variáveis como *factor* ou *numeric* e somente na última fazer modificar os labels;
- Nosso modelo está alinhado com o modelo teórico, como no artigo de referência(NASCIMENTO et al., 2003).

A circunferência abdominal tem uma relação maior com o diagnóstico de DM do que o IMC. O VLDL também tem elevada associação com o DM.

- Há um problema com desbalanceamento entre os gêneros (masculino e femino). Há muito mais mulheres com dianóstico de diabetes do que homens.

```
Na construção da árvore parece que somente o fato de ser homem já reduz a
possibilidade de DM.
```

Referências

DZIURA, J. D. et al. Strategies for dealing with missing data in clinical trials: from design to analysis. *The Yale journal of biology and medicine*, v. 86, n. 3, p. 343, 2013.

FORTI, A. et al. Diretrizes da Sociedade Brasileira de Diabetes 2019-2020 [Internet]. *São Paulo: Clannad*, 2020.

NASCIMENTO, R. do et al. Diabetes mellitus tipo 2: fatores preditivos na população nipo-brasileira. *Arquivos Brasileiros de Endocrinologia & Metabologia*, v. 47, n. 5, p. 584–592, 2003.