Modelagem Preditiva

Cesar Augusto Cavalheiro Marcondes
Matricula: 079149/2016
Turma: MSP016AP-ZEBABD- T2
Ano de Nascimento: 1976

Trabalho Individual do Curso Modelagem Preditiva (Prof. Abraham Laredo Sicsú)

MBA Executivo em Economia e Gestão: Business Analytics e Big Data
FGV Management São Paulo

1 Introdução

Para esse trabalho foi escolhido o Dataset chamado XZCALL. Esse Dataset está presente no arquivo "DADOS CURSO modelos preditivos 2017 03 03.xlsx". Dessa planilha foi extraída por copy/paste somente a aba XZCALL em um arquivo csv separado. Também foram substituídos os acentos das palavras do Dataset para uma versão "sem acentos". O resultado é o arquivo **"xzcall.csv"**.

Conforme solicitação do professor, omitimos neste relatório, produzido usando *R Markdown*, todas as referencias as chamadas em código do R para os gráficos. Portanto, somente mostraremos os resultados dessas execuções. Entretanto, os modelos e alguns códigos especiais (ex. imputação por k-NN) podem apresentar diferenças de um gabarito predefinido, portanto, iremos apresentar os comandos quando conveniente. O restante do código Rmd pode ser acessado no github - https://github.com/cmarcond/laredo-t1.git

2 Análise e Limpeza da Variáveis do Dataset

Conforme pode ser verificado a seguir,o Dataset tem 3368 observações dispostas em 9 variáveis. A descrição completa do Dataset está no livro "LIVRO REGRESSÃO LOGISTICA 2016 09 19.pdf" na página 69. O status é a variável resposta que a empresa gostaria de predizer, corresponde aos "bons" funcionários que permaneceram na empresa nos 12 meses completos, e os maus ficaram menos tempo.

```
## 'data.frame':    3368 obs. of  9 variables:
##  $ FUNCIONARIO: chr  "X1001" "X1002" "X1003" "X1004" ...
##  $ STATUS     : chr  "mau" "bom" "bom" "mau" ...
##  $ IDADE      : chr  "25 - 45" "25 - 45" "maior que 45" "25 - 45" ...
##  $ ECIV       : chr  "casado" "casado" "viuvo" "casado" ...
##  $ DIST_EMP   : chr  "media" "proximo" "media" "proximo" ...
##  $ TIPORESID  : chr  "propria" "propria" "propria" "propria" ...
##  $ PRIM_EMP   : chr  "nao" "nao" "nao" "nao" ...
##  $ TESTE      : int  82 72 35 80 84 78 83 94 84 65 ...
##  $ EDUC       : chr  "superior" "secundario" "secundario" "secundario" ...
```

2.1 Analisar as diferentes variáveis previsoras (corrigir outliers e missing values)

Iniciamos verificando a incidencia de outliers com o R. A conclusão é que não encontramos nenhum campo em branco no Dataset, como mostra o resultado abaixo.

```
table(is.na(xzcall))
```

```
##
## FALSE
## 30312
```

2.1.1 Missing Values e Imputação por k-NN

Entretanto, estudando os níveis das variáveis categóricas, descobrimos que todas elas tem níveis que fazem sentido, exceto TIPORESID. A variável TIPORESID possui alguns empregados com o valor "3" nesse campo, ao invés de residencia "própria" ou "outros". O valor "3" pode ter sido um erro de digitação, ou poderia estar representando um valor vazio dependendo do dicionário de dados. Como não temos a definição, vamos assumir que esse valor seja equivalente a "missing value". Usaremos o método de imputação automática por k-NN (com k = 10, valor default, embora testamos com outros k, sem diferença). A tabela TIPORESID antes da imputação possui 30 valores "3", que serão substituídos por NAs. Esses dados representam 0.89 % do total de individuos.

```
table(TIPORESID)
```

```
## TIPORESID
##       3  outros propria
##      30     223    3115
```

```
## TIPORESID
##  outros propria    <NA>
##     223    3115      30
```

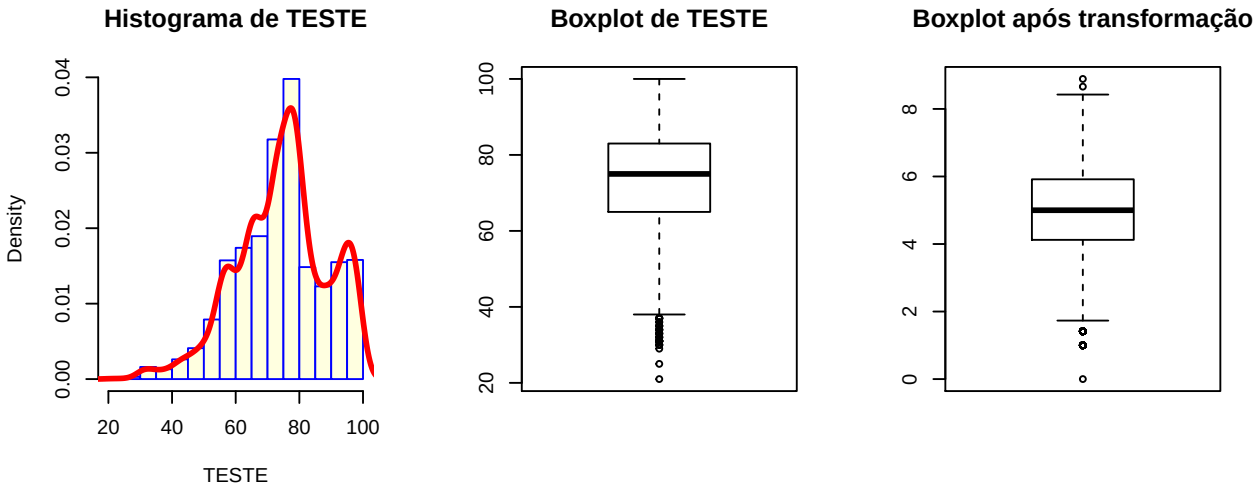
Usaremos o pacote **VIM - Visualization and Imputation of Missing Values**, que possui a função **kNN** que busca no Dataset por campos NA para substituição por valores de comportamento semelhante com os k-vizinhos mais próximos.

```
library(VIM)
xzcall <- kNN(xzcall, variable = c("TIPORESID"), k = 10)
table(xzcall$TIPORESID, useNA = "ifany")
```

```
##
##  outros propria
##     223    3145
```

O Dataset resultante não possui mais nenhum NA, e os valores imputados foram todos para o TIPORESID = "propria", conforme mostra a tabela abaixo. De fato, isso é coerente, com testes realizados com essa variável que indicam uma homogeneidade, por exemplo, a maioria dos individuos marcados com "3" tem idade > 45 e estão no seu primeiro emprego, e a grande maioria dos empregados tem o tipo de residencia, antes da imputação é do tipo "propria" com 92.4%.

2.1.2 Detecção de Outliers e Proposta de Transformação

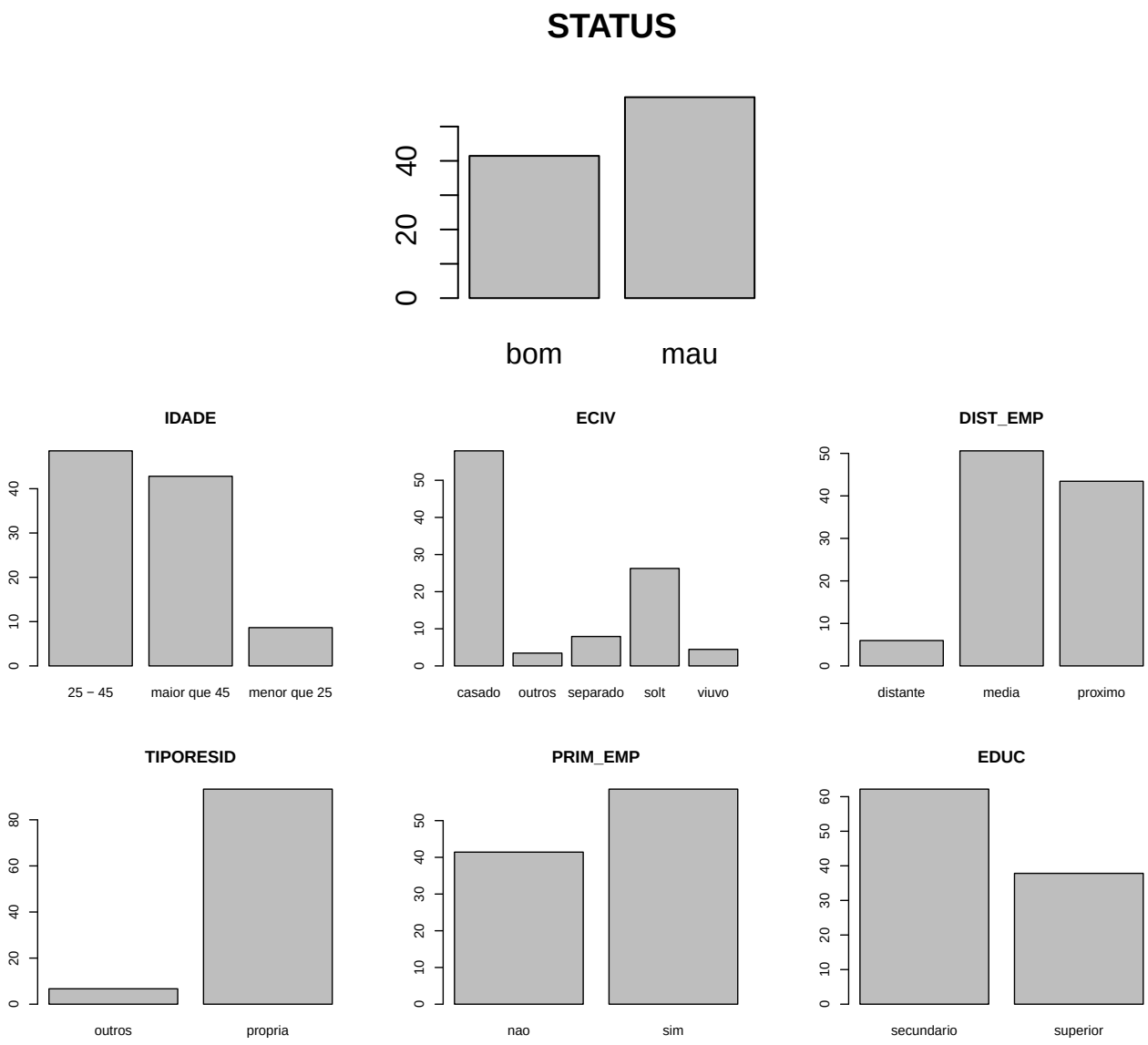


A única variável quantitativa é TESTE, as outras são variáveis categóricas. Portanto, vamos começar analisando os resultados de TESTE. O gráfico de histograma mostra que a distribuição dos resultados de teste, compreende no teste de admissão, mínimo de 21 e no máximo 100. A distribuição e o boxplot são levemente skewed negativamente (os gráficos à esquerda), desse modo foi aplicado uma transformação MAX(X) - SQRT(X), resultando em 5 outliers de 3368 pontos.

2.2 Analisando proporções das variáveis individuais

Analisando as proporções individuais das variáveis, por exemplo, é possível verificar se a variável resposta está bem representada no Dataset, ou se será necessário fazer técnicas para tratar desequilibrio entre valores da variável resposta, como ROSE ou outras. Desse modo, começando pela variável resposta (STATUS), verificamos equilibrio, e portanto não será necessário ajustes.

A seguir, nesta tabela abaixo, analisaremos as outras variáveis (todas convertidas em %).



2.3 Criação de Variáveis Dummy

A criação de dummy auxilia a melhorar a expressividade do modelo. Uma das variáveis que pode permitir dummies é a variável EDUC, que tem os valores "secundario" e "superior". Na verdade, note que o valor "superior" já pressupõe que a pessoa tenha se formado no "secundário". Portanto, podemos criar 2 variáveis DUMMY (BSEC e BSUP). Onde as seguintes clausulas lógicas são possíveis: "se secundário então bsec = 1 e bsup = 0", a outra opção é "se superior então bsec = 1 e bsup = 1". Realizamos esse procedimento e colocamos as novas variáveis no Dataset. A variável ECIV também tem uma estrutura semelhantes, pois

quem é viúvo ou separado, já foi casado, entretanto, o nível "outros" traz mais complexidade para uma transformação, pois não está claro se "outros" já foi casado, no passado, e agora está "amasiado".

2.4 Estrutura final do Dataset após procedimentos de limpeza

Para finalizar a preparação do Dataset, além dos campos novos de TESTESQRT, TIPORESID imputado, BSEC e BSUP também excluímos as variáveis que não precisamos, como FUNCIONARIO, EDUC e mesmo TESTE. Para TESTE, note que, é possível transformar de volta TESTESQRT para TESTE apenas aplicando a inversa da transformação $max - x^2$, exemplo $4.24^2 = 17.9776; 100 - 17.9776 = 82$. Dessa forma, a estrutura final do Dataset para treinamento e testes é esse:

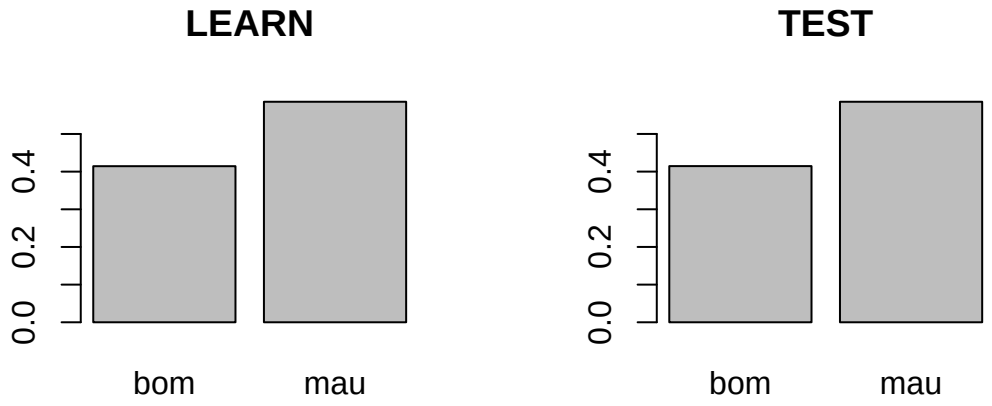
```
## 'data.frame':    3368 obs. of  9 variables:
##  $ STATUS   : Factor w/ 2 levels "bom","mau": 2 1 1 2 1 1 1 2 1 2 ...
##  $ IDADE    : Factor w/ 3 levels "25 - 45","maior que 45",..: 1 1 2 1 2 1 1 1 1 2 ...
##  $ ECIV     : Factor w/ 5 levels "casado","outros",..: 1 1 5 1 2 1 4 4 1 1 ...
##  $ DIST_EMP : Factor w/ 3 levels "distante","media",..: 2 3 2 3 3 2 3 2 3 3 ...
##  $ TIPORESID: Factor w/ 2 levels "outros","propria": 2 2 2 2 2 2 2 2 2 2 ...
##  $ PRIM_EMP : Factor w/ 2 levels "nao","sim": 1 1 1 1 1 1 1 2 1 2 ...
##  $ TESTESQRT: num  4.24 5.29 8.06 4.47 4 ...
##  $ BSEC     : Factor w/ 1 level "1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ BSUP     : Factor w/ 2 levels "0","1": 2 1 1 1 1 2 1 2 1 2 ...
```

3 Divisão do Dataset em 2 grupos (learn e test)

Realizamos a parte A) do trabalho após a fase de limpeza e criação de dummies porque o resultado de flags seria diferente antes e depois da limpeza. Portanto, dividindo o arquivo em 2 grupos, learn com 60% e test com 40% utilizando a função createdatapartition do pacote "caret". Para fins de validação, a soma dos flags gerados está listado abaixo, bem como verificamos a proporção de STATUS "bom" e "mau" em cada subconjunto (learn e test) permanece similar ao original (Seção 2.2).

```
sum(flag)
```

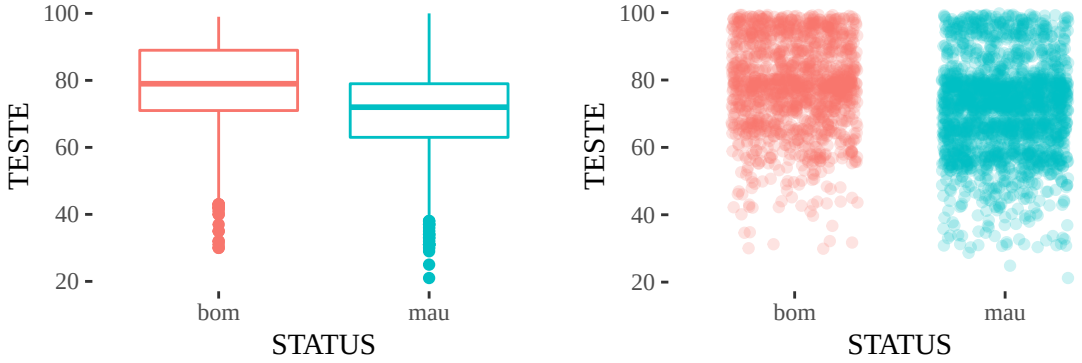
```
## [1] 3414848
```



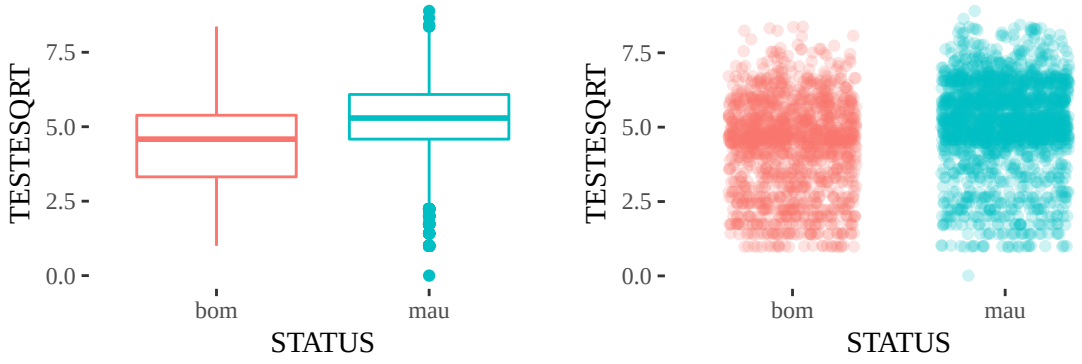
4 Poder discriminador das variáveis em relação ao alvo

4.1 TESTE - Previsora quantitativa

A única variável quantitativa é a nota do TESTE. Nessa seção, iremos comparar os valores da nota do TESTE versus o STATUS ser "bom" ou "mau". Algo interessante que notamos é que a variável TESTE, SEM a transformação pela raiz quadrada, tem o seu boxplot ligeiramente tendendo a associar melhores notas no teste à bons funcionários.



Entretanto, ao aplicar a transformação pela raiz quadrada (TESTESQRT), o boxplot fica o contrário, parece ligeramente tender a associar melhores notas à maus funcionários. Entretanto, ambos os boxplots são estatisticamente equivalentes, a mediana de cada um está dentro da área do IQR (Interquartile Range). Portanto, provavelmente, esta não é uma das primeiras variáveis mais importantes na predição.



4.2 DEMAIS VARIAVEIS - Previsoras qualitativas

Para as demais variáveis qualitativas iremos calcular o lambda de Goodman & Kruskal, e iremos comparar o poder preditivo de cada variável. Apresentaremos os resultados uma página para cada variável previsora. Lembrando que o parametro lambda representa perfeita associação quando o valor se aproxima de 1, e quando as variáveis qualitativa e alvo são independentes (ou seja, nenhuma relação entre elas) o valor se aproxima de 0. Além da interpretação visual dos dados, pela tabela de contingencia, também utilizamos uma implementação em R para extrair gamma de Goodman-Kruskal.

4.2.1 Variável IDADE vs STATUS

Criamos a tabela de contingencia da variável IDADE em relação a variável alvo STATUS. É possível visualmente ver alguma diferença, especialmente quando a idade é maior que 45, quando a proporção (em relação a todos os pontos) de "mau" é o dobro de "bom", nessa faixa (29% para "mau" versus 14% para "bom"). O valor de gamma de Goodman-Kruskal é $\lambda = 0.19$. Ou seja, **não** tem uma boa relação.

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  3368
##
##
##              | xzcall$STATUS
## xzcall$IDADE |       bom |       mau | Row Total |
## -------------|-----------|-----------|-----------|
##      25 - 45 |       791 |       844 |      1635 |
##              |      0.23 |      0.25 |           |
## -------------|-----------|-----------|-----------|
## maior que 45 |       457 |       985 |      1442 |
##              |      0.14 |      0.29 |           |
## -------------|-----------|-----------|-----------|
## menor que 25 |       148 |       143 |       291 |
##              |      0.04 |      0.04 |           |
## -------------|-----------|-----------|-----------|
## Column Total |      1396 |      1972 |      3368 |
## -------------|-----------|-----------|-----------|
##
##
```

4.2.2 Variável ECIV e STATUS

Criamos a tabela de contingencia da variável ECIV em relação a variável alvo STATUS. É possível visualmente ver alguma diferença, especialmente quando o funcionário é casado, quando a proporção (em relação a todos os pontos) de "mau" é um pouco maior que "bom", nessa faixa (36% para "mau" versus 22% para "bom"). O valor de gamma de Goodman-Kruskal é $\lambda = 0.1$. Ou seja, **não** tem uma boa relação.

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  3368
##
##
##              | xzcall$STATUS
##  xzcall$ECIV |       bom |       mau | Row Total |
## -------------|-----------|-----------|-----------|
##       casado |       749 |      1202 |      1951 |
##              |      0.22 |      0.36 |           |
## -------------|-----------|-----------|-----------|
##       outros |        55 |        61 |       116 |
##              |      0.02 |      0.02 |           |
## -------------|-----------|-----------|-----------|
##     separado |       125 |       142 |       267 |
##              |      0.04 |      0.04 |           |
## -------------|-----------|-----------|-----------|
##         solt |       428 |       456 |       884 |
##              |      0.13 |      0.14 |           |
## -------------|-----------|-----------|-----------|
##        viuvo |        39 |       111 |       150 |
##              |      0.01 |      0.03 |           |
## -------------|-----------|-----------|-----------|
## Column Total |      1396 |      1972 |      3368 |
## -------------|-----------|-----------|-----------|
##
##
```

4.2.3 Variável DIST_EMP e STATUS

Criamos a tabela de contingencia da variável DIST_EMP em relação a variável alvo STATUS. É possível visualmente ver alguma diferença, especialmente quando o funcionário mora próximo do trabalho, quando a proporção (em relação a todos os pontos) de "mau" é bem maior que "bom", nessa faixa (28% para "mau" versus 16% para "bom"). O valor de gamma de Goodman-Kruskal é $\lambda = 0.18$. Ou seja, **não** tem uma boa relação.

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  3368
##
##
##                  | xzcall$STATUS
## xzcall$DIST_EMP |       bom |       mau | Row Total |
## ----------------|-----------|-----------|-----------|
##        distante |       107 |        94 |       201 |
##                 |      0.03 |      0.03 |           |
## ----------------|-----------|-----------|-----------|
##           media |       753 |       951 |      1704 |
##                 |      0.22 |      0.28 |           |
## ----------------|-----------|-----------|-----------|
##         proximo |       536 |       927 |      1463 |
##                 |      0.16 |      0.28 |           |
## ----------------|-----------|-----------|-----------|
##    Column Total |      1396 |      1972 |      3368 |
## ----------------|-----------|-----------|-----------|
##
##
```

4.2.4 Variável TIPO_RESID e STATUS

Criamos a tabela de contingencia da variável TIPORESID em relação a variável alvo STATUS. É possível visualmente ver pouquissima diferença. A principal diferença é 41% e 55% (todos os pontos) - residencia própria, que é equivalente a razão dessas variáveis no Dataset. O valor de gamma de Goodman-Kruskal é $\lambda = 0.02$. Ou seja, praticamente 0, indicando independencia. Basta notar que TIPORESID no Dataset, mais de 90% dos pontos são residencia própria.

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  3368
##
##
##                   | xzcall$STATUS
## xzcall$TIPORESID |       bom |       mau | Row Total |
## -----------------|-----------|-----------|-----------|
##           outros |        94 |       129 |       223 |
##                  |      0.03 |      0.04 |           |
## -----------------|-----------|-----------|-----------|
##          propria |      1302 |      1843 |      3145 |
##                  |      0.39 |      0.55 |           |
## -----------------|-----------|-----------|-----------|
##     Column Total |      1396 |      1972 |      3368 |
## -----------------|-----------|-----------|-----------|
##
##
```

4.2.5 Variável PRIM_EMP e STATUS

Criamos a tabela de contingencia da variável PRIM_EMP em relação a variável alvo STATUS. É possível visualmente ver alguma diferença, especialmente quando o funcionário responde SIM, que é seu primeiro emprego, proporção (em relação a todos os pontos) de "mau" é bem maior que "bom" nessa faixa, quase 2.5 vezes (42% para "mau" versus 17% para "bom"). O valor de gamma de Goodman-Kruskal é $\lambda = 0.56$. Ou seja, **existe** uma BOA relação entre as variáveis PRIM_EMP e STATUS. Isso faz sentido porque em um trabalho de call-center, que é a maioria de primeiro emprego de muita gente, claramente, não é a opção ideal, e portanto, existe a tendencia em abandonar o emprego para outro melhor no futuro.

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  3368
##
##
##                  | xzcall$STATUS
## xzcall$PRIM_EMP |       bom |       mau | Row Total |
## ----------------|-----------|-----------|-----------|
##             nao |       822 |       573 |      1395 |
##                 |      0.24 |      0.17 |           |
## ----------------|-----------|-----------|-----------|
##             sim |       574 |      1399 |      1973 |
##                 |      0.17 |      0.42 |           |
## ----------------|-----------|-----------|-----------|
##    Column Total |      1396 |      1972 |      3368 |
## ----------------|-----------|-----------|-----------|
##
##
```

4.2.6 Variável EDUC e STATUS

Criamos a tabela de contingencia da variável EDUC em relação a variável alvo STATUS. É possível visualmente ver alguma diferença, especialmente quando o funcionário responde que possui diploma superior. A chance de ser um "mau" funcionário é 3.75 vezes proporcionalmente maior na faixa de ensino superior (com 30% para "mau" e 8% para "bom"). O valor de gamma de Goodman-Kruskal é $\lambda = 0.6$. Ou seja, valor próximo de 1, e portanto **existe** uma BOA relação.

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  3368
##
##
##              | xzcall$STATUS
##      OLDEDUC |       bom |       mau | Row Total |
## -------------|-----------|-----------|-----------|
##   secundario |      1116 |       978 |      2094 |
##              |      0.33 |      0.29 |           |
## -------------|-----------|-----------|-----------|
##     superior |       280 |       994 |      1274 |
##              |      0.08 |      0.30 |           |
## -------------|-----------|-----------|-----------|
## Column Total |      1396 |      1972 |      3368 |
## -------------|-----------|-----------|-----------|
##
##
```

4.3 Construcao da Árvore

A próxima etapa é a construção do modelo de Árvore de Decisão. Para esse modelo, usaremos a partição realizada anteriormente, e os dados de Learn serão usados para treinar o modelo, e os dados de Test serão usados para investigar o desempenho do modelo. Foi utilizado o algoritmo CART "Recursive partitioning" (rpart) e utilizadas todas as variáveis.

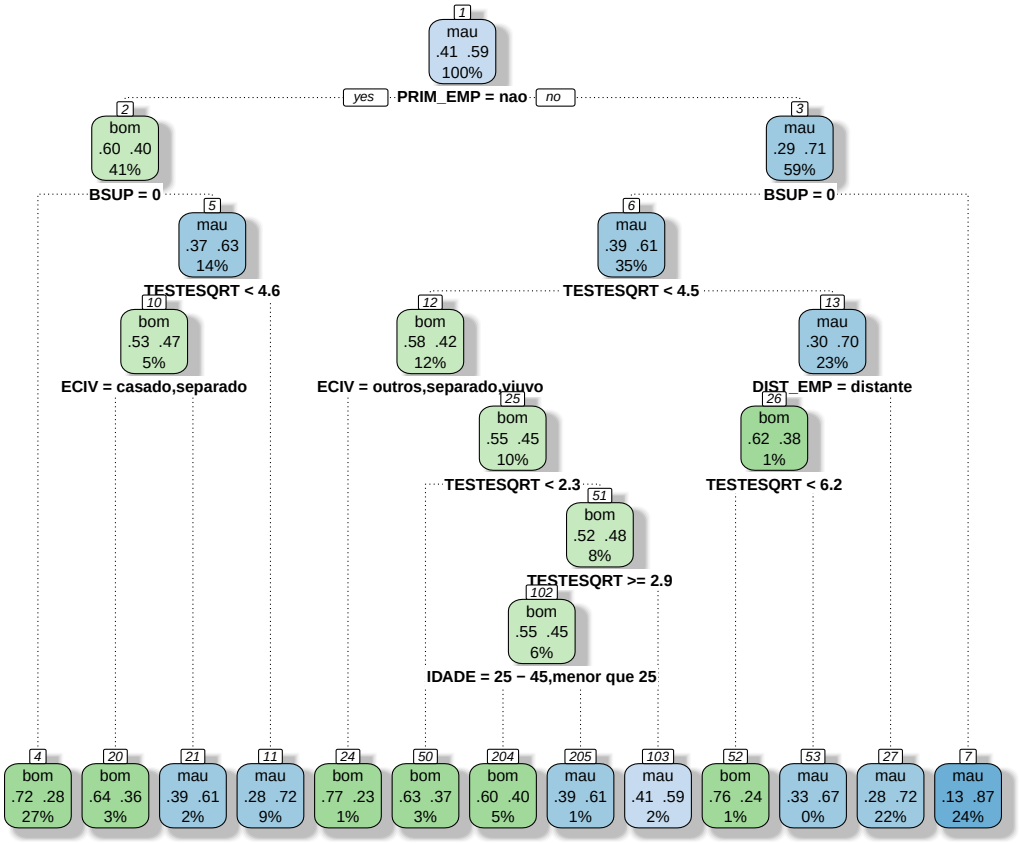
Nas primeiras execuções, notamos que a árvore pré-poda e a árvore pós-poda eram IGUAIS. Da maneira que os dados estão dispostos, nós notamos que o cálculo do risco e o ganho para o particionamento estava sendo afetado pelo parametro CP (complexity parameter) default de 0.01.

Aparentemente, o parametro CP limita o crescimento da árvore, porque os limiares dos valores de ganho que geram "splits" são de menor granularidade do que o CP = 0.01 permite. Fizemos alguns testes, e geramos novamente a árvore alterando o parametro de CP = 0.003 ao invés.

O resultado é uma árvore mais profunda e densa, como mostrado abaixo. Mostraremos o código para evitar confusões.

```
ad1=rpart(data=xzcall.learn, STATUS ~ IDADE + ECIV
          + DIST_EMP + TIPORESID + PRIM_EMP + TESTESQRT + BSEC + BSUP,
          method = "class", control = rpart.control(cp = 0.003))
```

4.4 Desenhando a Árvore de Decisão Pré-Poda



Notamos nessa árvore, que se for o primeiro emprego e se NÃO tiver curso superior, uma boa chance de ser "bom" funcionário. Essa árvore possui profundidade de 8 níveis e claramente esta super especializada (overfit).

4.5 Poda da árvore

Em seguida, fizemos o print do cptable que orienta a poda. Note que os intervalos entre CP começam a saltar de 0.003 em 0.003. O ponto de poda é baseado na metodologia do professor, onde ao invés de simplesmente usar o ponto de menor xerror que seria 0.67184, faz-se o ponto intermediário entre o CP anterior e esse do menor xerror. O procedimento foi automatizado com o código abaixo.

```
printcp(ad1)
```

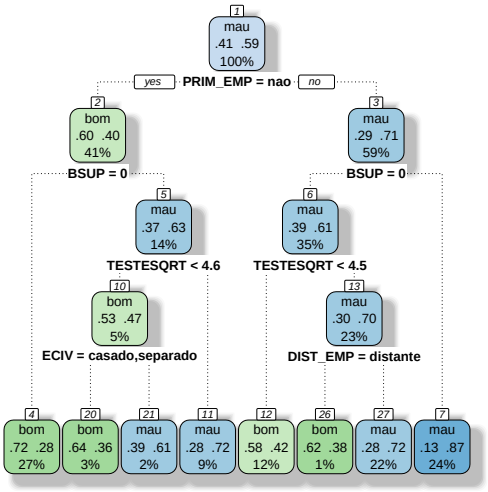
```
##
## Classification tree:
## rpart(formula = STATUS ~ IDADE + ECIV + DIST_EMP + TIPORESID +
##     PRIM_EMP + TESTESQRT + BSEC + BSUP, data = xzcall.learn,
##     method = "class", control = rpart.control(cp = 0.003))
##
## Variables actually used in tree construction:
## [1] BSUP      DIST_EMP  ECIV      IDADE     PRIM_EMP  TESTESQRT
##
## Root node error: 838/2022 = 0.41444
##
## n= 2022
##
##          CP nsplit rel error  xerror     xstd
## 1 0.1945107      0   1.00000 1.00000 0.026434
## 2 0.0883055      1   0.80549 0.80549 0.025305
## 3 0.0208831      2   0.71718 0.71838 0.024536
## 4 0.0095465      4   0.67542 0.68138 0.024156
## 5 0.0071599      6   0.65632 0.68138 0.024156
## 6 0.0035800      7   0.64916 0.67184 0.024052
## 7 0.0030000     12   0.63126 0.68974 0.024244
```

```
#
# indicando que o valor do CP correspondente é um valor de CP
# entre 0.0035800 e 0.0071599, portanto o meio desses é 0.005369928
index <- which.min(ad1$cptable[ , "xerror"])
tree_min <- (ad1$cptable[index - 1, "CP"] + ad1$cptable[index, "CP"])/2
tree_min
```

```
## [1] 0.005369928
```

```
ad2=prune(ad1, cp=tree_min)
```

4.6 Desenhando a Árvore de Decisão Pós-Poda



4.7 Print da Árvore após a poda

Para uma correta interpretação é preciso lembrar que TESTESQRT é a variável transformada, basta alterar o valor por $100 - X^2$. Portanto, 4.6 é equivalente a uma nota de 78.84.

```
## n= 2022
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##  1) root 2022 838 mau (0.4144411 0.5855589)
##    2) PRIM_EMP=nao 837 337 bom (0.5973716 0.4026284)
##      4) BSUP=0 551 157 bom (0.7150635 0.2849365) *
##      5) BSUP=1 286 106 mau (0.3706294 0.6293706)
##       10) TESTESQRT< 4.636496 104  49 bom (0.5288462 0.4711538)
##         20) ECIV=casado,separado 58  21 bom (0.6379310 0.3620690) *
##         21) ECIV=outros,solt,viuvo 46  18 mau (0.3913043 0.6086957) *
##       11) TESTESQRT>=4.636496 182  51 mau (0.2802198 0.7197802) *
##    3) PRIM_EMP=sim 1185 338 mau (0.2852321 0.7147679)
##      6) BSUP=0 708 277 mau (0.3912429 0.6087571)
##       12) TESTESQRT< 4.527356 233  99 bom (0.5751073 0.4248927) *
##       13) TESTESQRT>=4.527356 475 143 mau (0.3010526 0.6989474)
##         26) DIST_EMP=distante 26  10 bom (0.6153846 0.3846154) *
##         27) DIST_EMP=media,proximo 449 127 mau (0.2828508 0.7171492) *
##      7) BSUP=1 477  61 mau (0.1278826 0.8721174) *
```

5 Análise de Desempenho do Modelo Árvore de Decisão

5.1 Importância das variáveis no processo de construção da árvore

Analisando a importancia das variáveis é possível perceber que o modelo parece coerente com a nossa análise prévia das variáveis previsoras. Por exemplo, PRIMEIRO EMPREGO, CURSO SUPERIOR e, eventualmente, o valor da nota do TESTE são as variáveis com maior importancia no modelo.

```
##  PRIM_EMP      BSUP TESTESQRT     IDADE      ECIV  DIST_EMP
##    95.585    84.205    35.096     9.116     8.099     5.503
```

5.2 Estimação das probabilidades e classificação com corte default (0.5)

```
prob2= predict(ad2, newdata = xzcall.test, type = "prob")
clas2= predict(ad2, newdata = xzcall.test, type = "class")
```

5.3 Matriz de Classificação colocando a classificação REAL na linha, e a prevista na coluna

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  1346
##
##
##                     | clas2
## xzcall.test$STATUS |       bom |       mau | Row Total |
## -------------------|-----------|-----------|-----------|
##                bom |       370 |       188 |       558 |
##                    |      0.27 |      0.14 |           |
## -------------------|-----------|-----------|-----------|
##                mau |       202 |       586 |       788 |
##                    |      0.15 |      0.44 |           |
## -------------------|-----------|-----------|-----------|
##       Column Total |       572 |       774 |      1346 |
## -------------------|-----------|-----------|-----------|
##
##
```

5.4 Acurácia do Modelo

A acurácia do modelo calculada pela diagonal da matriz de confusão dividido pelo número de pontos de teste foi igual a 71%.

5.5 Indicadores do poder discriminador da árvore

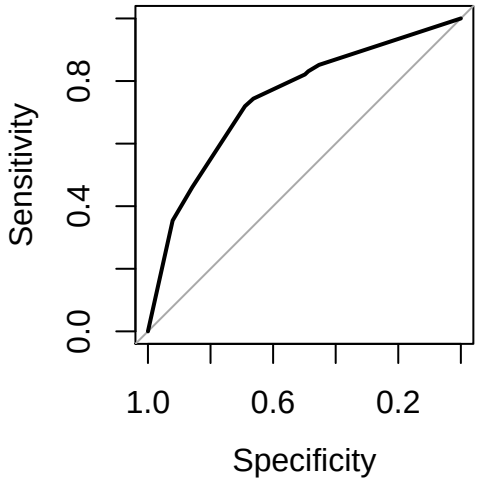
Outros indicadores foram calculados para a árvore pós-poda.

```
## Class labels have been switched from (bom,mau) to (0,1)
```

```
##            H Gini   AUC  AUCH   KS  MER   MWL Spec.Sens95 Sens.Spec95   ER
## scores 0.215 0.49 0.745 0.746 0.41 0.29 0.287       0.301        0.13 0.29
##         Sens  Spec Precision Recall   TPR   FPR    F Youden  TP  FP  TN
## scores 0.744 0.663     0.757  0.744 0.744 0.337 0.75  0.407 586 188 370
##         FN
## scores 202
```

5.6 Curva ROC Pós-Poda

Outros indicadores, como curva ROC, foram calculados para a árvore pós-poda.




```
## Area under the curve: 0.7449
```

5.7 Comparação da proporção da variável TESTE do Dataset com as variáveis TESTE prevista pelo modelo

```
library(arules)
knota = discretize(xzcall.test$TESTE, method = 'frequency', categories = 10)
table(knota, xzcall.test$STATUS)
```

```
##
## knota         bom mau
##   [0.00,2.45)  89  62
##   [2.45,3.61)  74  58
##   [3.61,4.58)  97  66
##   [4.58,4.80)  69  62
##   [4.80,5.00)  42  56
##   [5.00,5.39)  52 123
##   [5.39,5.83)  39  79
##   [5.83,6.16)  30  92
##   [6.16,6.71)  38  95
##   [6.71,8.37]  28  95
```

```
table(knota, clas2)
```

```
##              clas2
## knota         bom mau
##   [0.00,2.45) 103  48
##   [2.45,3.61) 105  27
##   [3.61,4.58) 128  35
##   [4.58,4.80)  46  85
##   [4.80,5.00)  32  66
##   [5.00,5.39)  47 128
##   [5.39,5.83)  37  81
##   [5.83,6.16)  29  93
##   [6.16,6.71)  29 104
##   [6.71,8.37]  16 107
```

5.8 Comparação da proporção das probabilidades que acertam o valor de bom, mal pelo modelo

```
kprob = discretize(prob2[ ,1], method = 'frequency', categories = 4)
table(kprob, xzcall.test$STATUS)
```

```
##
## kprob           bom mau
##   [0.128,0.283)  79 363
##   0.283          94 204
##   [0.391,  Inf] 385 221
```

```
table(kprob, clas2)
```

```
##               clas2
## kprob          bom mau
##   [0.128,0.283)   0 442
##   0.283           0 298
##   [0.391,  Inf] 572  34
```

6 Sumário

Tomamos a liberdade de realizar todos os comentários, mais importantes, ao longo do trabalho. Portanto, nosso sumário final é bem enxuto. Em resumo, o modelo apresentou uma acurácia abaixo da expectativa de aprox. 71%. A manipulação de variáveis alterou pouco desse resultado. Um possível estudo futuro seria usar outros modelos como regressão logistica e florestas aleatórias. O estudo de cada uma das variáveis mostrou uma forma interessante de entender e explicar os resultados do modelo. Testamos pouco a parte de parametrização, como minimo de folhas e número mínimo para "split", e isso poderia ser melhorado em futuros trabalhos.