Lista 05

Matheus Cougias e Klysman Rezende

31/08/2020

Leitura do arquivo

Realiza a leitura do arquivo em formato .csv, onde as 9 primeiras variáveis são valores decimais e a variável resposta (y) é binária. Assim, foi necessário fazer a alteração dessa variável para binário, pois o R a considerou como valores inteiros de 0 ou 1. Também foram necessárias alterações nas demais variáveis, já que elas não foram consideradas como factor, e sim como números.

```
dados <- read.csv("cancer.csv")
dados$y <- as.factor(dados$y)
```

Análise inicial

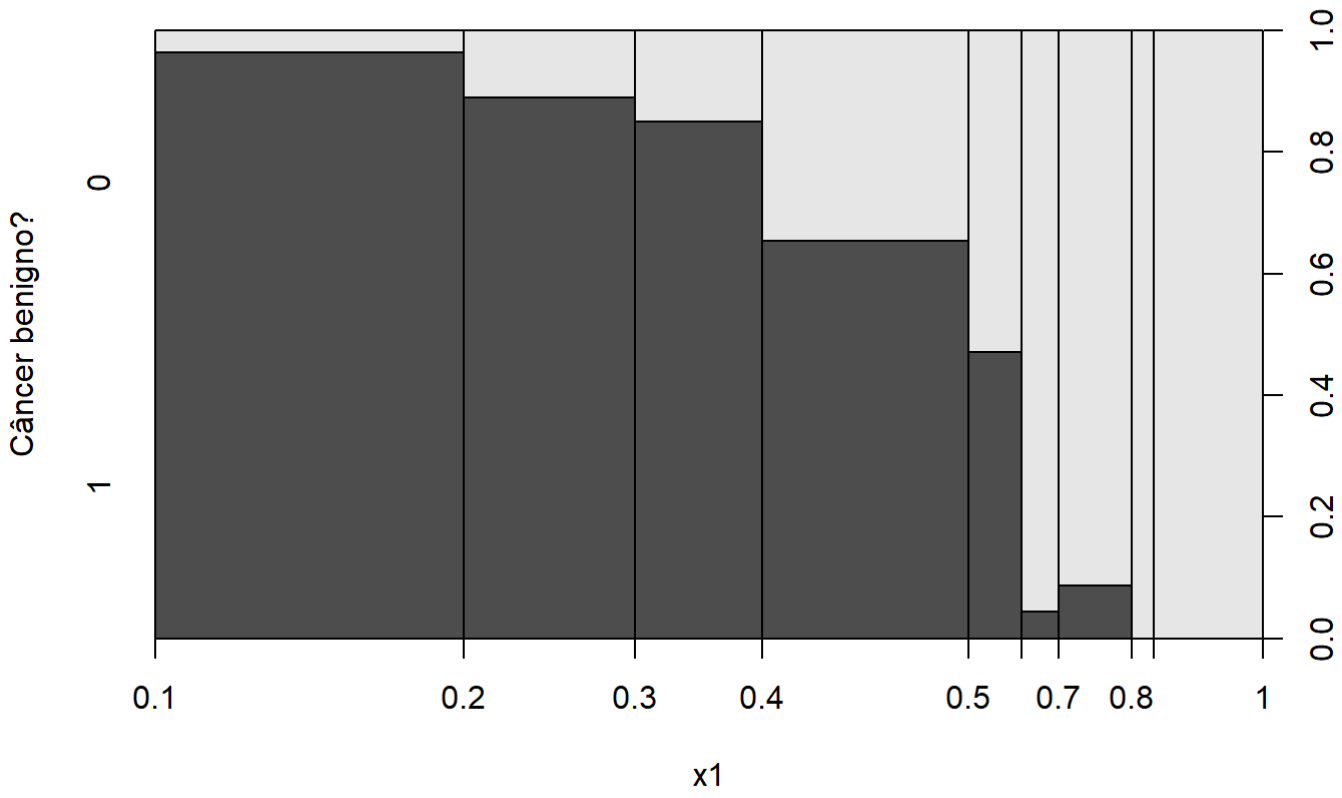
A partir do carregamento dos dados, já é possíve realizar uma análise inicial que os dados apresentam. A primeira característica dos dados nos mostra que a proporção de câncer benigno estudado foi cerca de duas vezes maior que a de malignos (65,52% e 34,38% respectivamente). Assim, busca-se identificar o perfil que apresenta melhor probabilidade do câncer ser benigno, comparando os valores apresentados para cada variável.

Sabendo disso, foi realizada a montagem de gráficos que comparam cada intervalo das variáveis (x1, x1, …, x9) com a proporção de câncer benigno para aquele intervalo. Em todas as variáveis foi apontado que o valor que possui essa maior proporção está no intervalo 0,1 e 0,2. Por outro lado, o intervalo que apresenta maior proporção de câncer maligno foi o de 0.8 e 1 para quase todas as 9 variáveis estudadas.

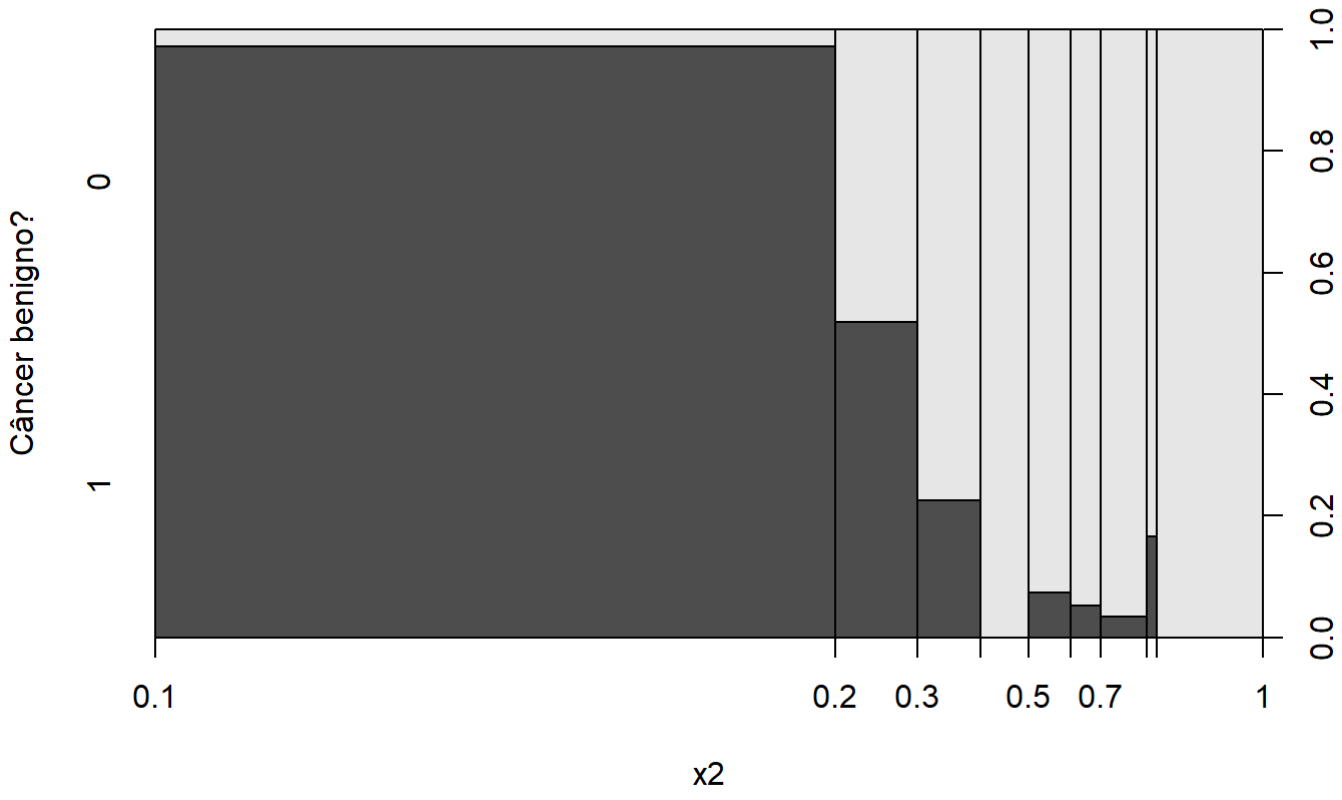
```
table(dados$y)/sum( table(dados$y))
```

```
##
##         0         1
## 0.3447783 0.6552217
```

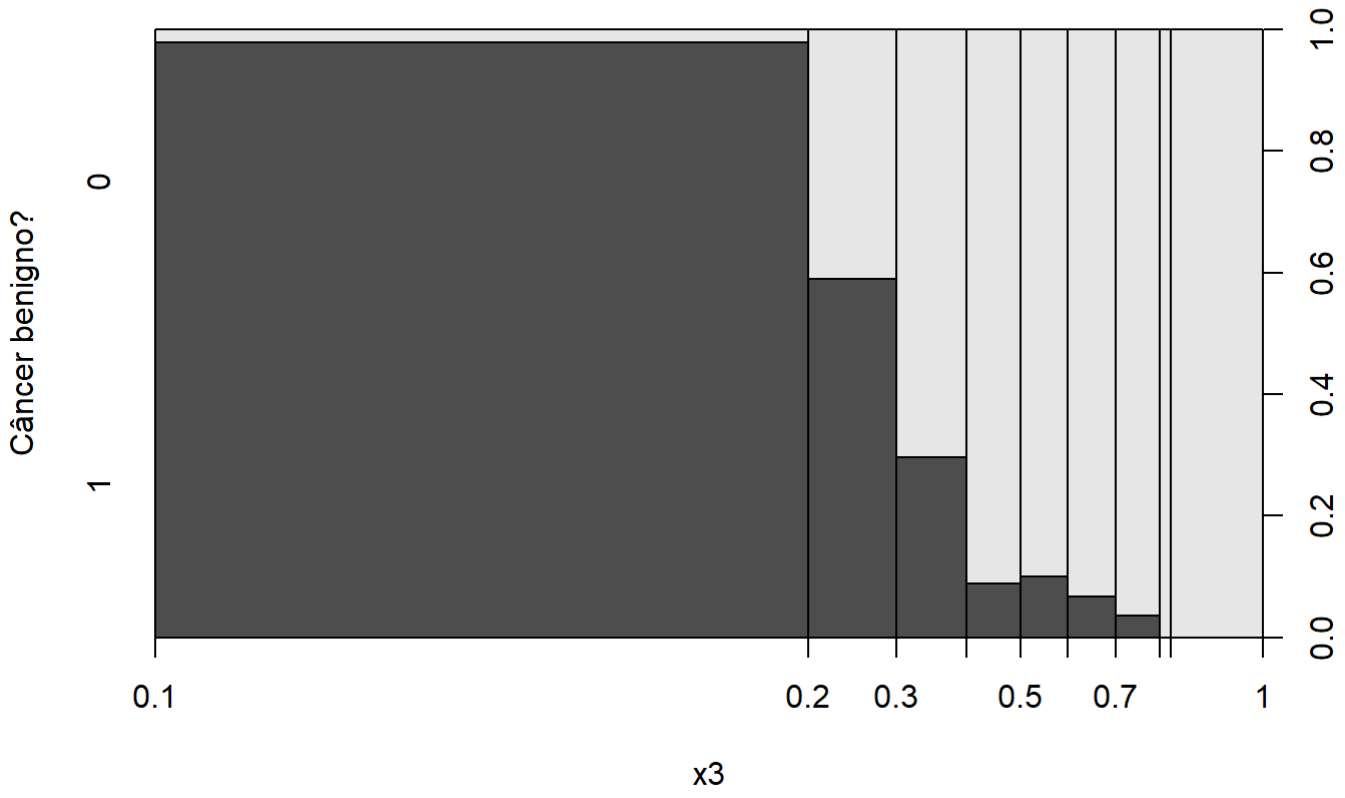
```
plot(y ~ x1, data = dados, ylab="Câncer benigno?")
```



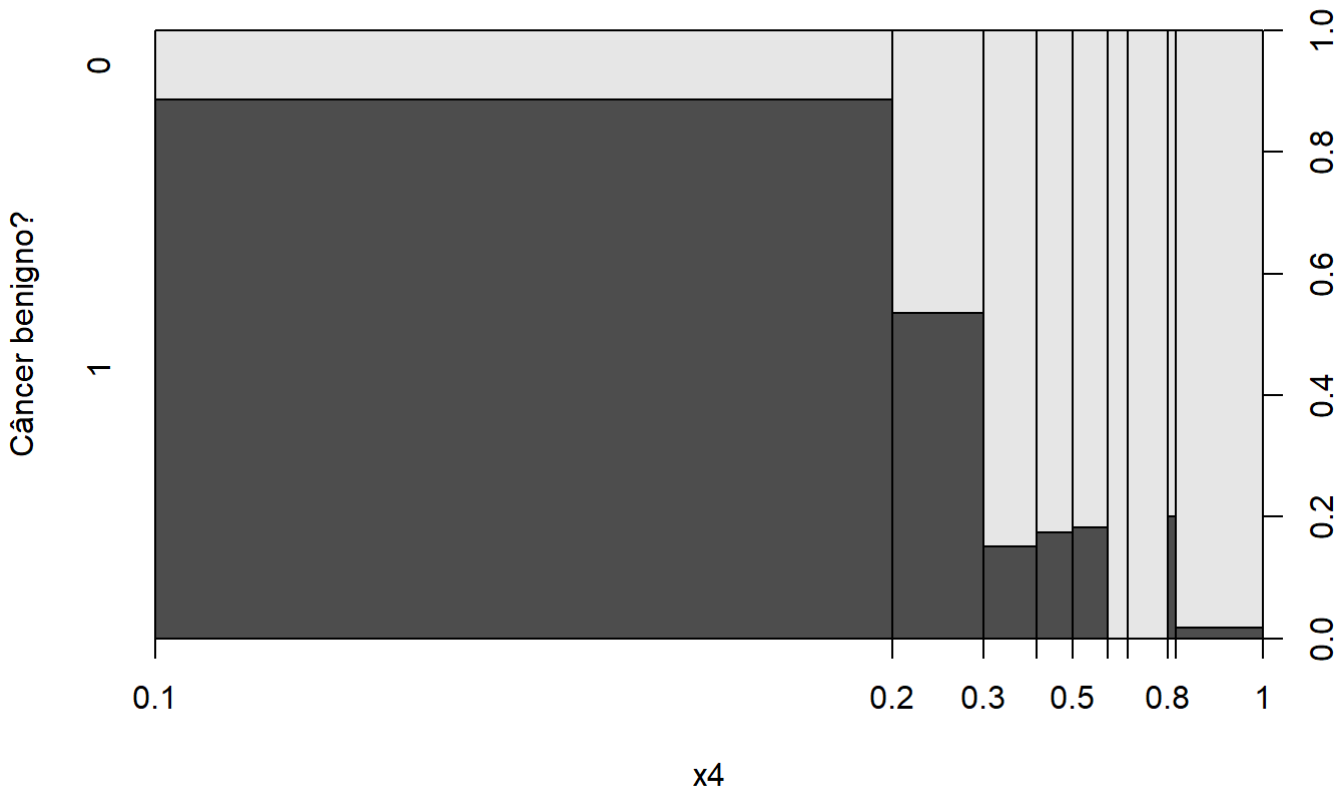

```
plot(y ~ x2, data = dados, ylab="Câncer benigno?")
```



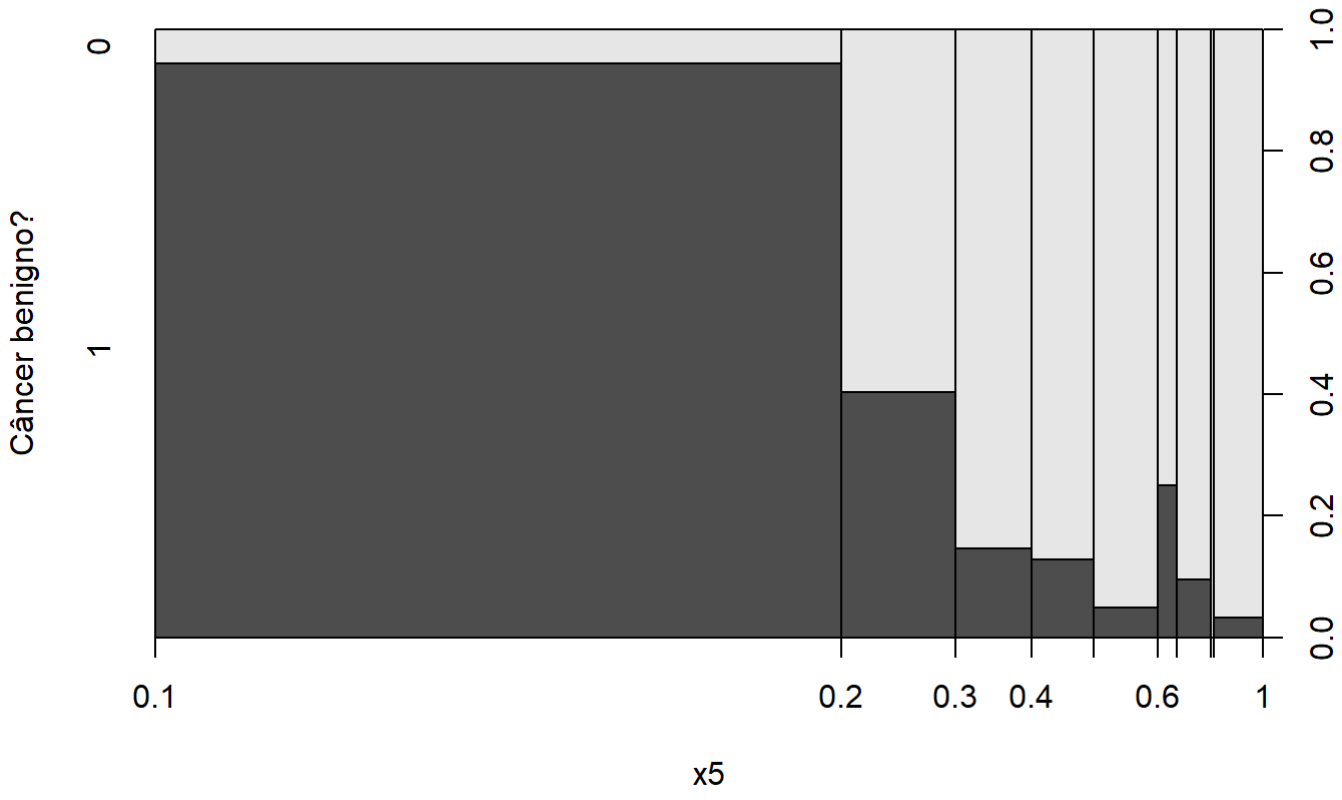
```
plot(y ~ x3, data = dados, ylab="Câncer benigno?")
```



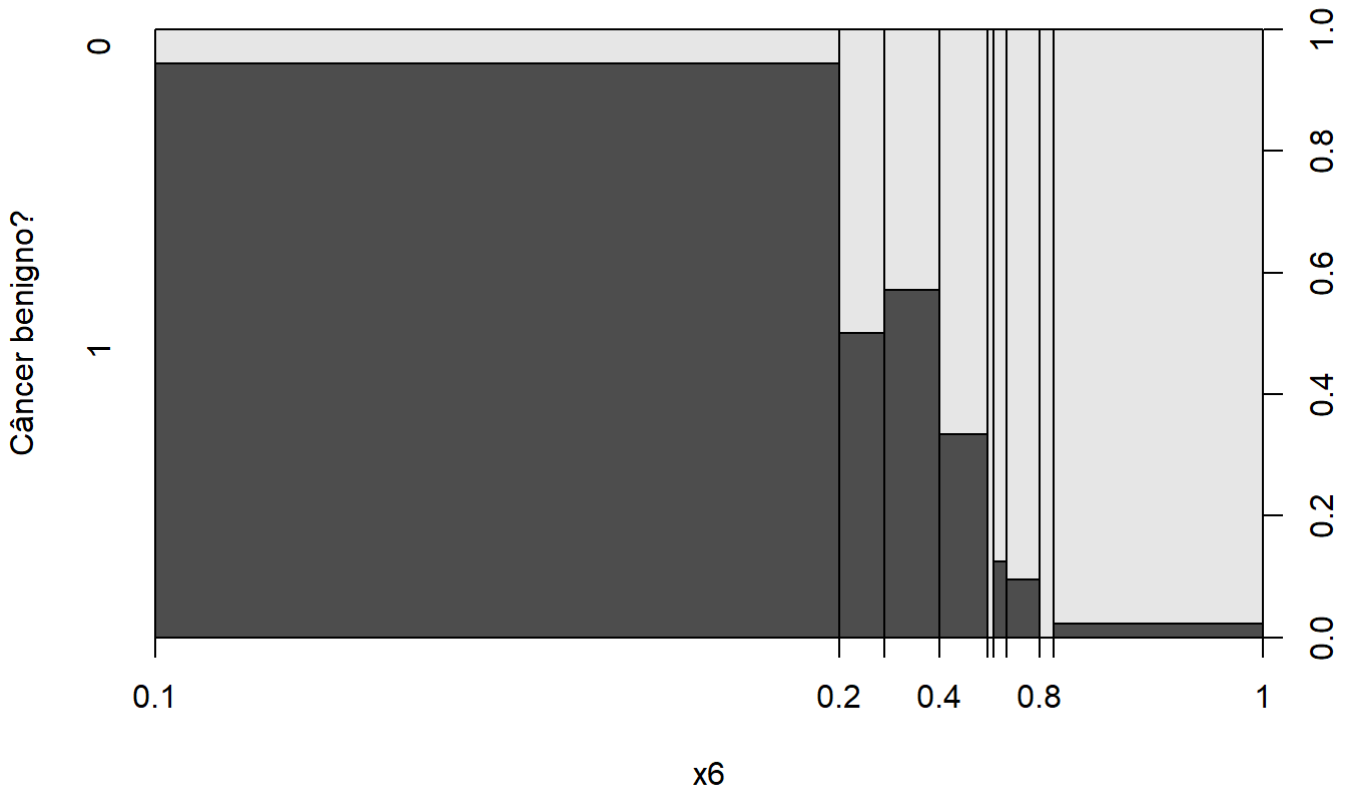

```
plot(y ~ x4, data = dados, ylab="Câncer benigno?")
```



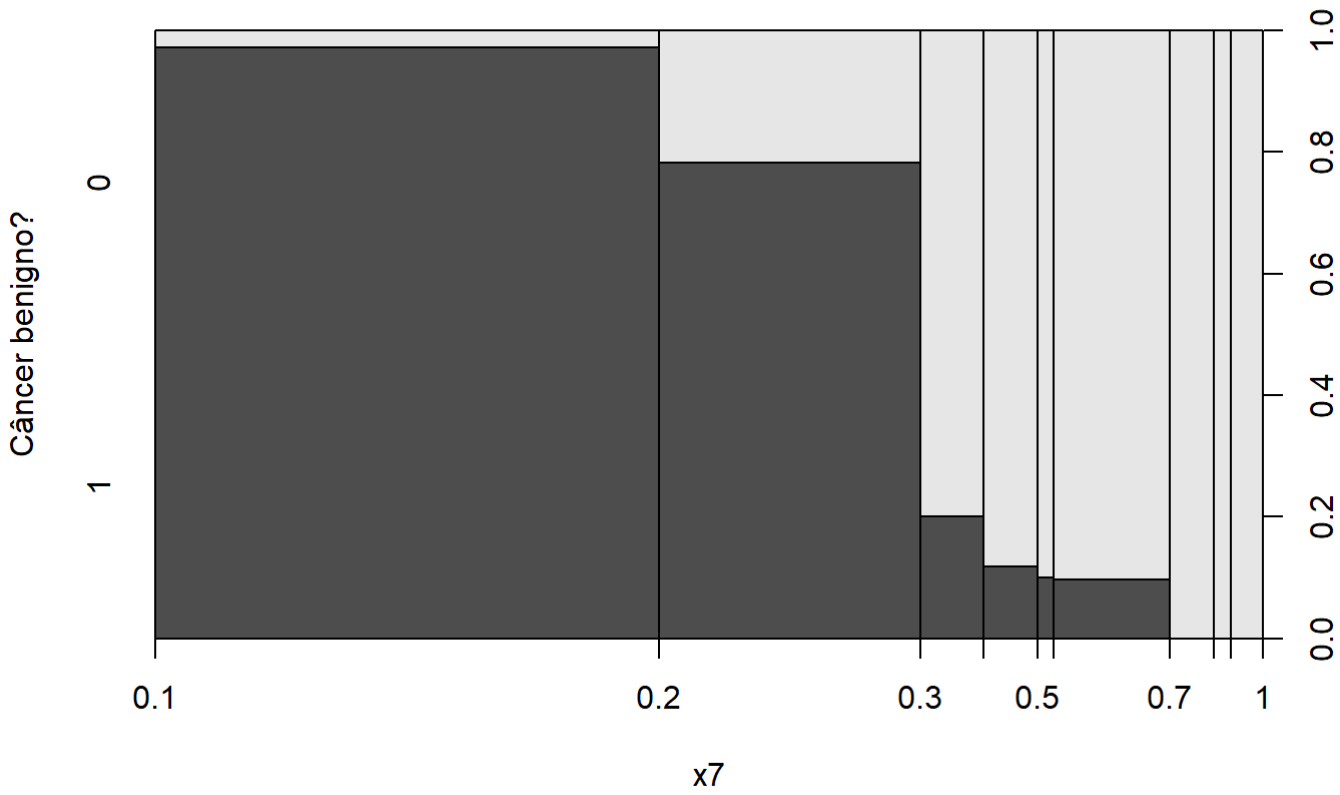

```
plot(y ~ x5, data = dados, ylab="Câncer benigno?")
```



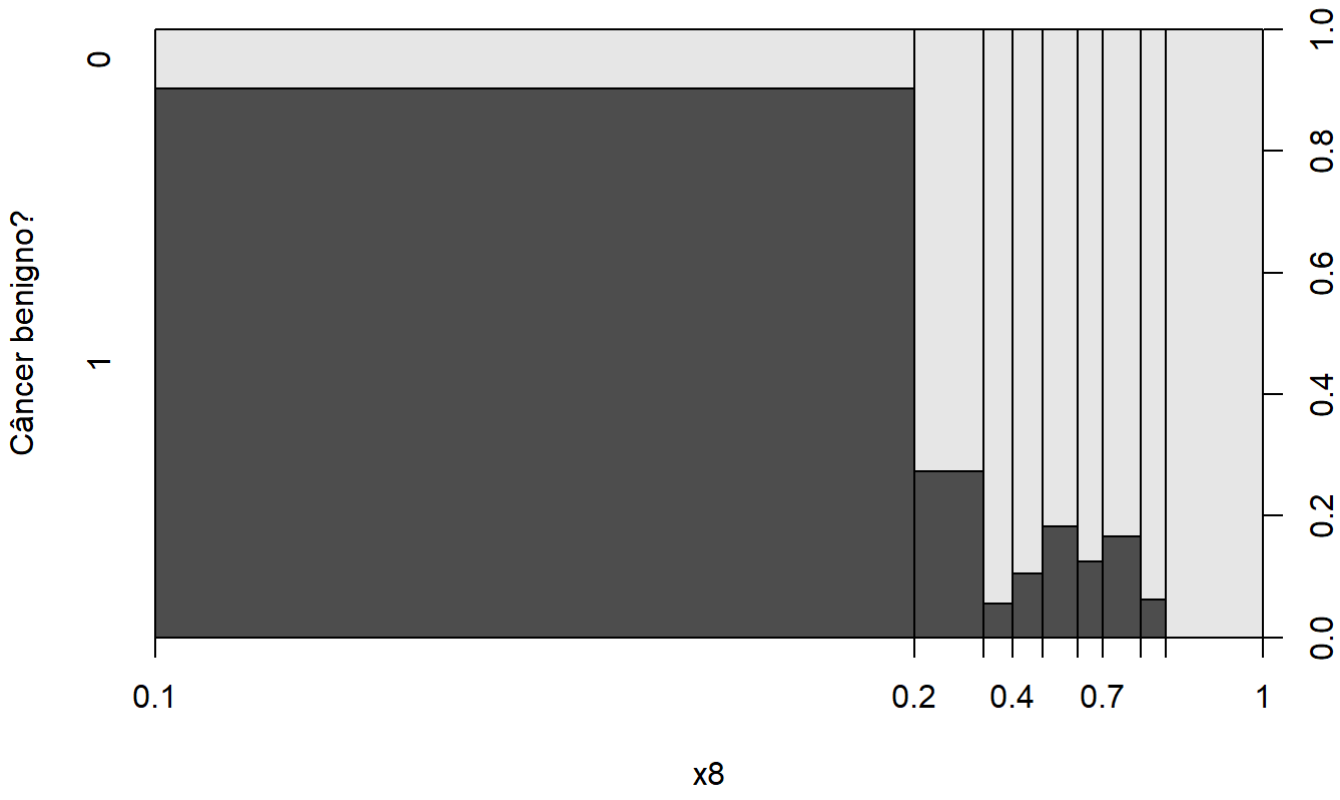
```
plot(y ~ x6, data = dados, ylab="Câncer benigno?")
```



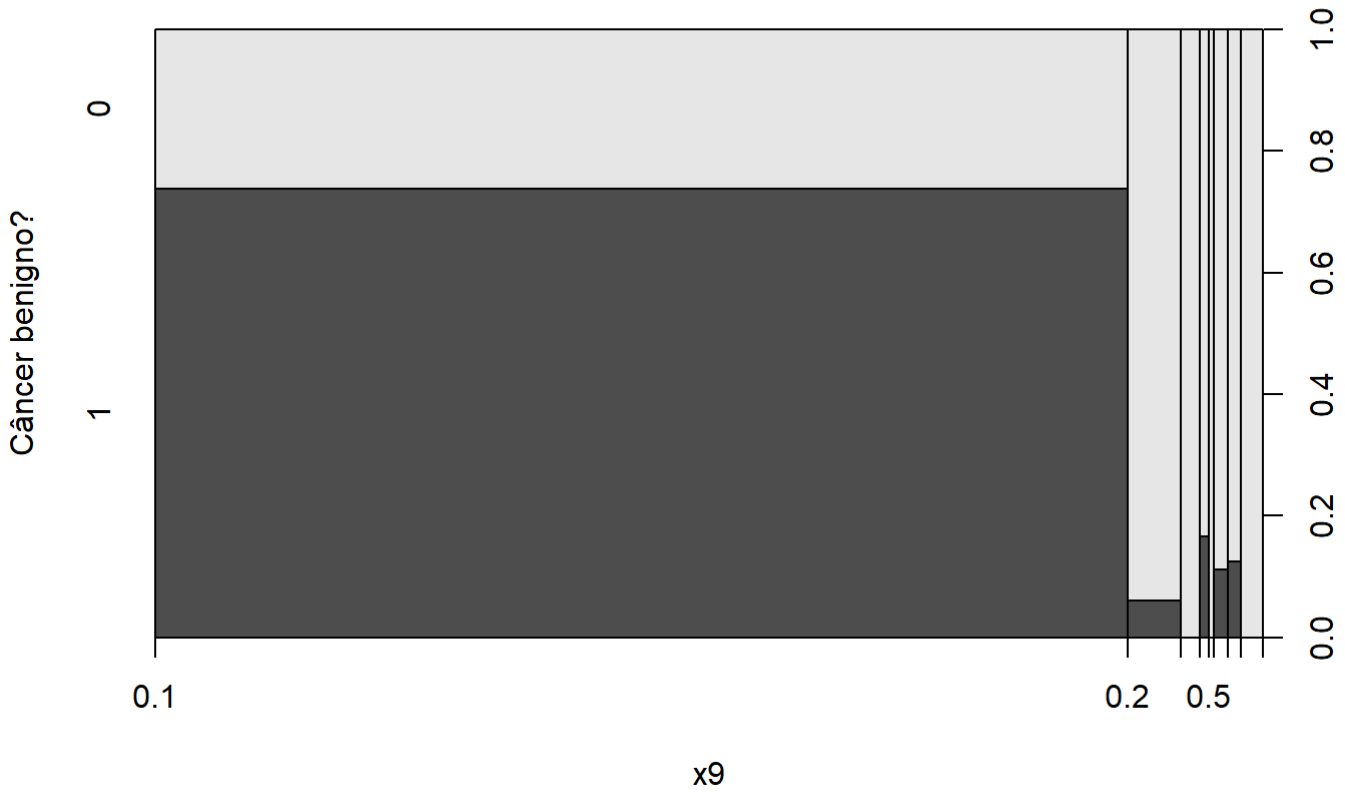

```
plot(y ~ x7, data = dados, ylab="Câncer benigno?")
```




```
plot(y ~ x8, data = dados, ylab="Câncer benigno?")
```



```
plot(y ~ x9, data = dados, ylab="Câncer benigno?")
```



Ajuste modelo logistico univariado

Dessa maneira, já com a ideia em mente de quais valores provavelmente serão utilizados para traçar o perfil desejado, um modelo de regressão logístia é aplicado ao problema, desingnando realmente quais serão os intervalos adotados. Lembrando que nesse primeiro teste todas as variáveis serão utilizadas, sem a seleção de quais realmente são relevantes para o problema.

A regressão logística identifica que todas as variáveis fazem com que a tendência do câncer gerado seja maligno, quanto maior for o seu valor, por estarem descritas como variáveis contínuas (numéricas). Isso demonstra que a análise efetuada no último tópico é verdadeira, onde para que o valor resultante seja câncer benigno, TODAS as variáveis devem estar próximas do valor zero (que no caso, o único possível é o 0,1).

As variáveis que mais afetam a variação do valor de y são aquelas com o maior valor de Estimate (ou então menor valor de Pr(>|z|)). Neste caso, as variáveis mais relevantes pro problema são: x1, x4, x6 e x7.

```
modelo <- glm(y ~ ., family="binomial", data=dados)
summary(modelo)
```

```
##
## Call:
## glm(formula = y ~ ., family = "binomial", data = dados)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4210  -0.0292   0.0669   0.1284   3.2802
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  9.67304    1.05484   9.170  < 2e-16 ***
## x1          -5.31275    1.32332  -4.015 5.95e-05 ***
## x2          -0.06979    1.87499  -0.037  0.97031
## x3          -3.29874    2.08615  -1.581  0.11382
## x4          -2.39190    1.15245  -2.075  0.03794 *
## x5          -0.67396    1.51202  -0.446  0.65579
## x6          -4.07364    0.90035  -4.525 6.05e-06 ***
## x7          -4.09211    1.56267  -2.619  0.00883 **
## x8          -1.46488    1.02509  -1.429  0.15300
## x9          -5.48645    3.02780  -1.812  0.06998 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 900.53  on 698  degrees of freedom
## Residual deviance: 116.25  on 689  degrees of freedom
## AIC: 136.25
##
## Number of Fisher Scoring iterations: 8
```

Ajuste utilizando a função step()

Considerando que algumas variáveis podem predizer o valor das outras, podemos aplicar a função step() para diminuir o tamanho do problema, ou seja, fazer com que o número de variáveis analisadas diminua. Dessa maneira, as variáveis que foram mantidas ao final dos passos do step foram x1, x3, x4, x6, x7, x8 e x9. Todas as variáveis tidas como relevantes para o problema no tópico anterior foram mantidas e continuam sendo as que mais possuem o poder de alterar o valor final de y.

```
modelo <- glm(y ~ ., family="binomial", data=dados)
modelo <- step(modelo)
```

```
## Start:  AIC=136.25
## y ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9
##
##        Df Deviance    AIC
## - x2    1   116.26 134.26
## - x5    1   116.45 134.45
## <none>      116.25 136.25
## - x8    1   118.34 136.34
## - x3    1   118.62 136.62
## - x4    1   120.57 138.57
## - x9    1   120.76 138.76
## - x7    1   123.64 141.63
## - x1    1   136.41 154.41
## - x6    1   140.67 158.67
##
## Step:  AIC=134.26
## y ~ x1 + x3 + x4 + x5 + x6 + x7 + x8 + x9
##
##        Df Deviance    AIC
## - x5    1   116.46 132.46
## <none>      116.26 134.26
## - x8    1   118.44 134.44
## - x4    1   120.75 136.75
## - x9    1   120.90 136.90
## - x3    1   121.03 137.03
## - x7    1   124.03 140.03
## - x1    1   136.99 152.99
## - x6    1   140.67 156.67
##
## Step:  AIC=132.46
## y ~ x1 + x3 + x4 + x6 + x7 + x8 + x9
##
##        Df Deviance    AIC
## <none>      116.46 132.46
## - x8    1   119.04 133.04
## - x9    1   121.36 135.36
## - x4    1   121.84 135.84
## - x3    1   121.95 135.95
## - x7    1   124.53 138.53
## - x1    1   137.43 151.43
## - x6    1   141.55 155.55
```

```
summary(modelo)
```

```
##
## Call:
## glm(formula = y ~ x1 + x3 + x4 + x6 + x7 + x8 + x9, family = "binomial",
##     data = dados)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3789  -0.0278   0.0669   0.1280   3.3142
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   9.6109     1.0165   9.455  < 2e-16 ***
## x1           -5.3392     1.3137  -4.064 4.82e-05 ***
## x3           -3.5135     1.6060  -2.188  0.02869 *
## x4           -2.5146     1.0995  -2.287  0.02220 *
## x6           -4.1115     0.8964  -4.587 4.51e-06 ***
## x7           -4.1792     1.5403  -2.713  0.00666 **
## x8           -1.5712     0.9942  -1.580  0.11405
## x9           -5.5204     2.9839  -1.850  0.06430 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 900.53  on 698  degrees of freedom
## Residual deviance: 116.46  on 691  degrees of freedom
## AIC: 132.46
##
## Number of Fisher Scoring iterations: 8
```

```
valor = 604.00
probStep = exp(valor)/(1+exp(valor))
```