MAD – Data Analysis & Biostatistics in R

Logistic Regression

James R. Hunter, Ph.D.

DIPA, EPM, UNIFESP

9 October 2020


MAD

Section 1

Logistic Regression

Extension of Basic Regression Concepts

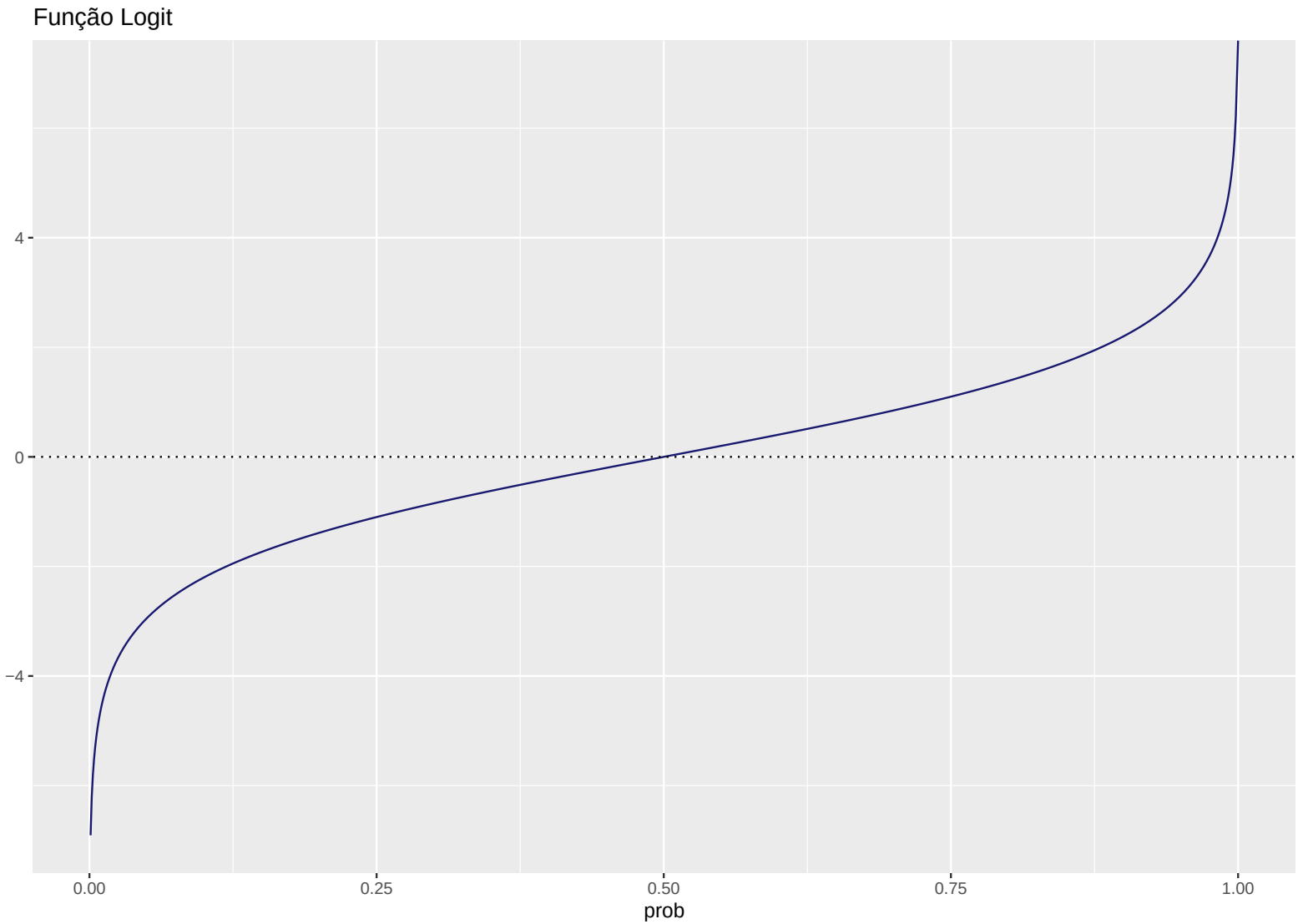
- Used frequently in biostatistics
- Variable Y is now a **binomial** variable
  - Only has 2 states:
    - TRUE; FALSE
    - 1;0
    - R5; X4
    - Infected; Not Infected
- As with SLR and MLR, covariates can be numeric or categorical

`logit` Function

- *log-odds*
- *odds* of an event
  - Probability of an event occurring divided by the probability of it not occurring
- **logit** natural logarithm of the odds

$$logit(p) = \frac{p}{1-p}$$

Logit Function

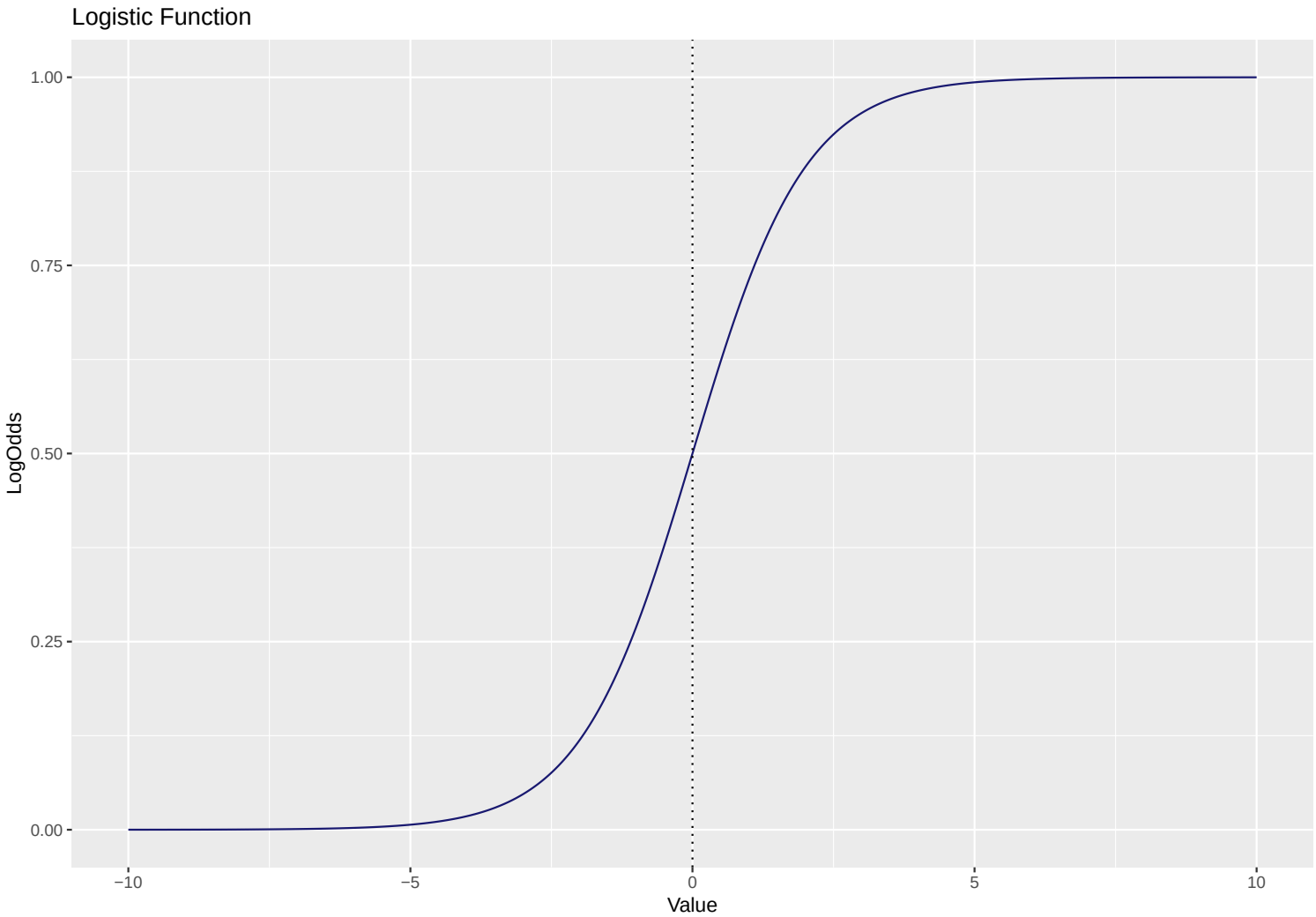


Logistic Function

- Function applied to independent variables (X)
  - Result: Dependent variable stays in interval between 0 and 1
    - ★ Range of probabilities
- **Logistic** function
- Inverse of the **logit**
- Can be applied to any number

$$logit^{-1}(x) = \frac{1}{1 + e^{-x}}$$

Logistic Function Graph



Compare SLR with Logistic Regression

- Linear Regression (using matrix notation)

$$y = X\beta + \epsilon_i$$

- Logistic Regression

$$p(y_i = 1) = logit^{-1}(X_i\beta) + \epsilon_i$$

General Linear Models

- Logistic regression prime example of class of models: general linear model (GLM)
  - A special case of GLM
- They manipulate the matrices differently than do the SLR models
- Other GLM models: poisson (count data)
- Output will be similar to the SLR output

Example: Patients with Coronary Heart Disease (CHD)

- Study of 100 patients
- Relation between the patient's age and CHD
- Data comes fromHosmer & Lemeshow, *Applied Logistic Regression* (2a Ed.)
  - File: `chdage.csv`

Load the Data

```
chdage <- read_csv(here::here("chdage.csv")) %>%
  mutate(chd = factor(chd)) %>%
  mutate(chd = fct_recode(chd, negativo = "0", positivo = "1"))
glimpse(chdage)
```

```
## Rows: 100
## Columns: 3
## $ id    <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18...
## $ idade <dbl> 20, 23, 24, 25, 25, 26, 26, 28, 28, 29, 30, 30, 30, 30, 30, 3...
## $ chd   <fct> negativo, negativo, negativo, negativo, positivo, negativo, n...
```

Basic Exploratory Analysis

```
chdage %>%
  select(idade) %>%
  descr(transpose = TRUE,
        stats = c("mean", "sd", "min", "q1", "med", "q3",
                  "max", "iqr", "cv"))
```

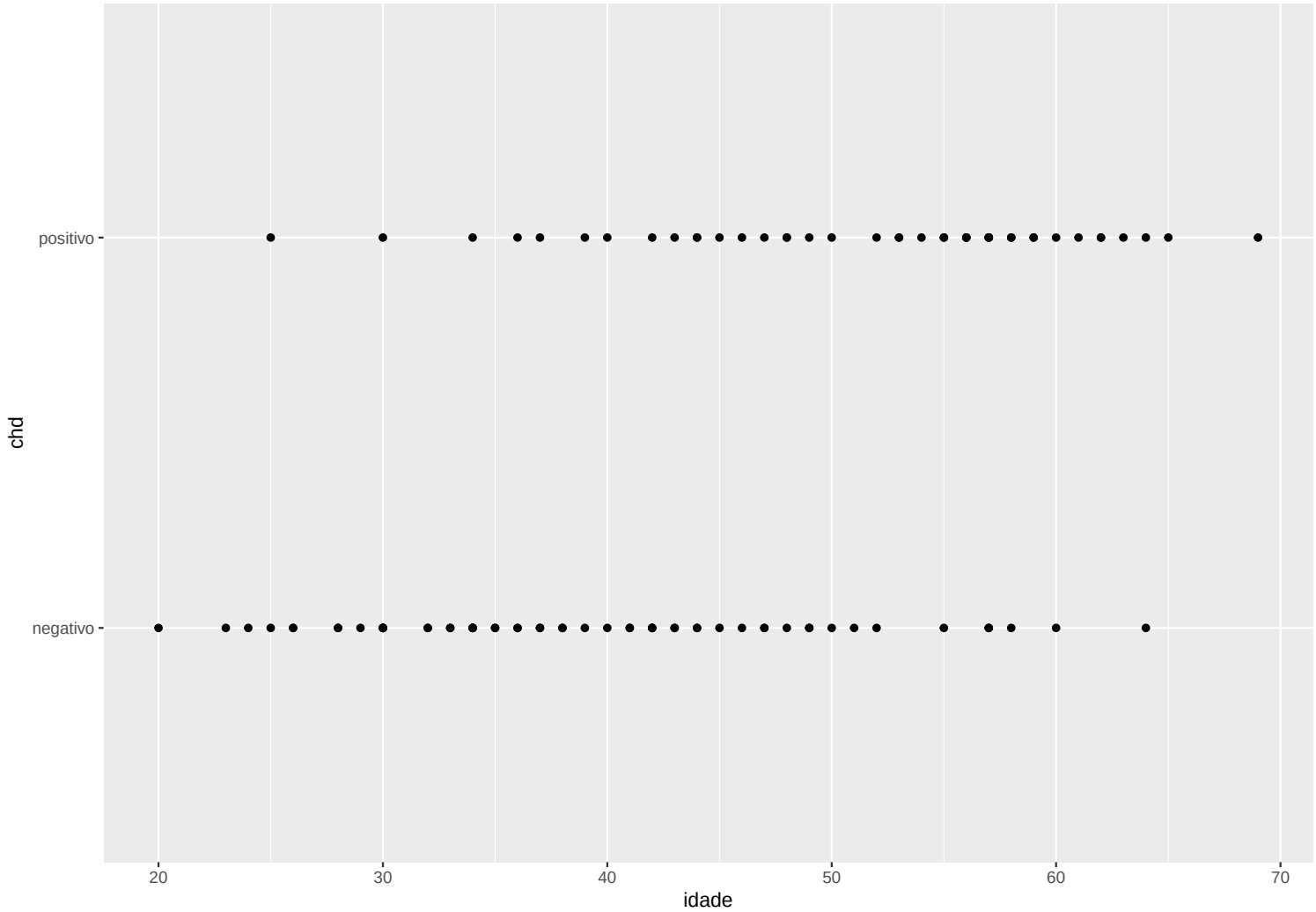
```
## Descriptive Statistics
## chdage$idade
## N: 100
##
##                Mean   Std.Dev      Min       Q1   Median       Q3      Max      IQR      CV
## ----------- ------- --------- ------- ------- -------- ------- ------- ------- ------
##       idade   44.38     11.72   20.00   34.50    44.00   55.00   69.00   20.25   0.26
```

```
chdage %>%
  select(chd) %>%
  freq()
```

```
## Frequencies
## chdage$chd
## Type: Factor
##
##                 Freq   % Valid   % Valid Cum.   % Total   % Total Cum.
## -------------- ------ --------- -------------- --------- --------------
##       negativo     57     57.00          57.00     57.00          57.00
##       positivo     43     43.00         100.00     43.00         100.00
##           <NA>      0                               0.00         100.00
##          Total    100    100.00         100.00    100.00         100.00
```

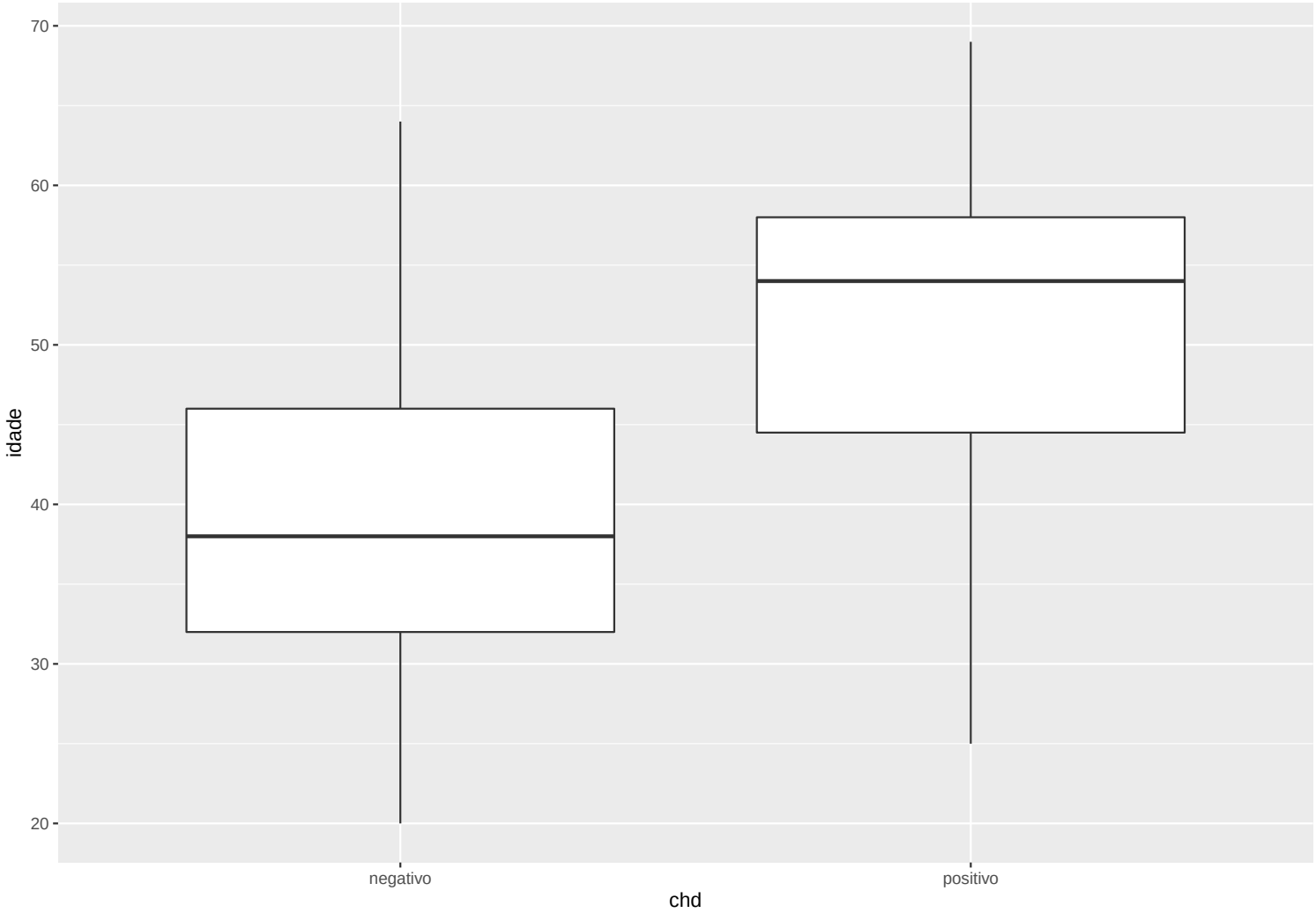
DotPlot of CHD x Idade

```
chdscat <- ggplot(data = chdage, aes(y = chd, x = idade)) + geom_point()
chdscat
```



Boxplot of Age

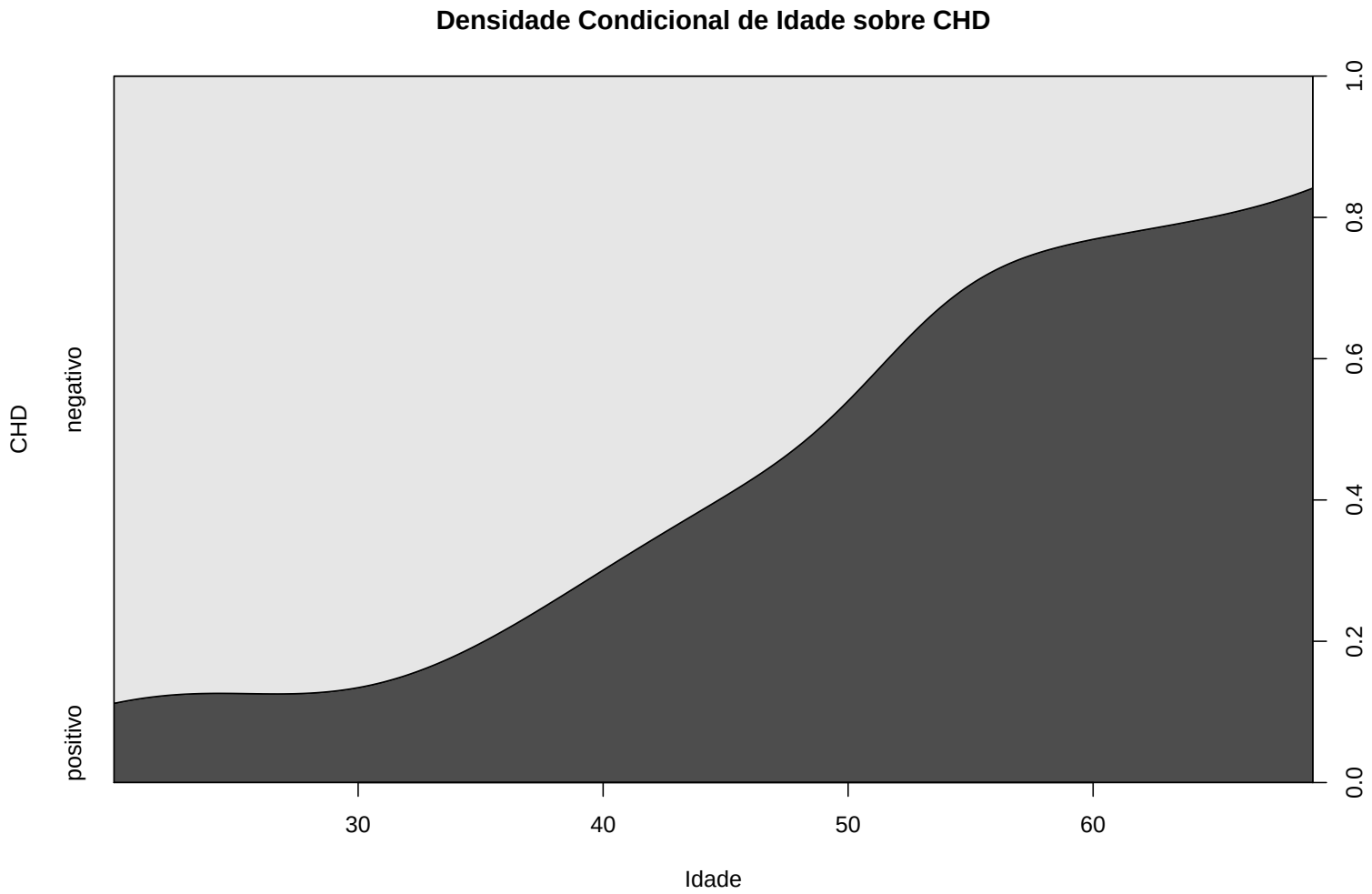
```
chdbox <- ggplot(data = chdage, aes(x = chd, y = idade, group = chd))
chdbox <- chdbox + geom_boxplot()
chdbox
```



Plot of Conditional Density

- Also useful for understanding how age changes with the 2 categories of CHD
- Shows the number with CHD (chd = 1) for all ages
  - As if `chd` were continuous
- Function `cdplot()` is in base R

```r
cdplot(factor(chd) ~ idade, data = chdage,
       main = "Densidade Condicional de Idade sobre CHD",
       xlab = "Idade", ylab = "CHD")
```



Model

- Like function `lm`, `glm` uses the formula format to specify the model
  - Dependent variable ~ independent variables
  - Independent variables separated by +
- Where the data come from (`data =`)
- `Family` of the model (in this case, `binomial`)
- Link function (in this case, `logit`)

```
chdfit1 <- glm(chd ~  idade, data = chdage,
               family = binomial(link = "logit"))
```

Results

- Use `summary()` to get results (as with `lm()`)
- Graph to review results with `coefplot()`
  - In package with same name

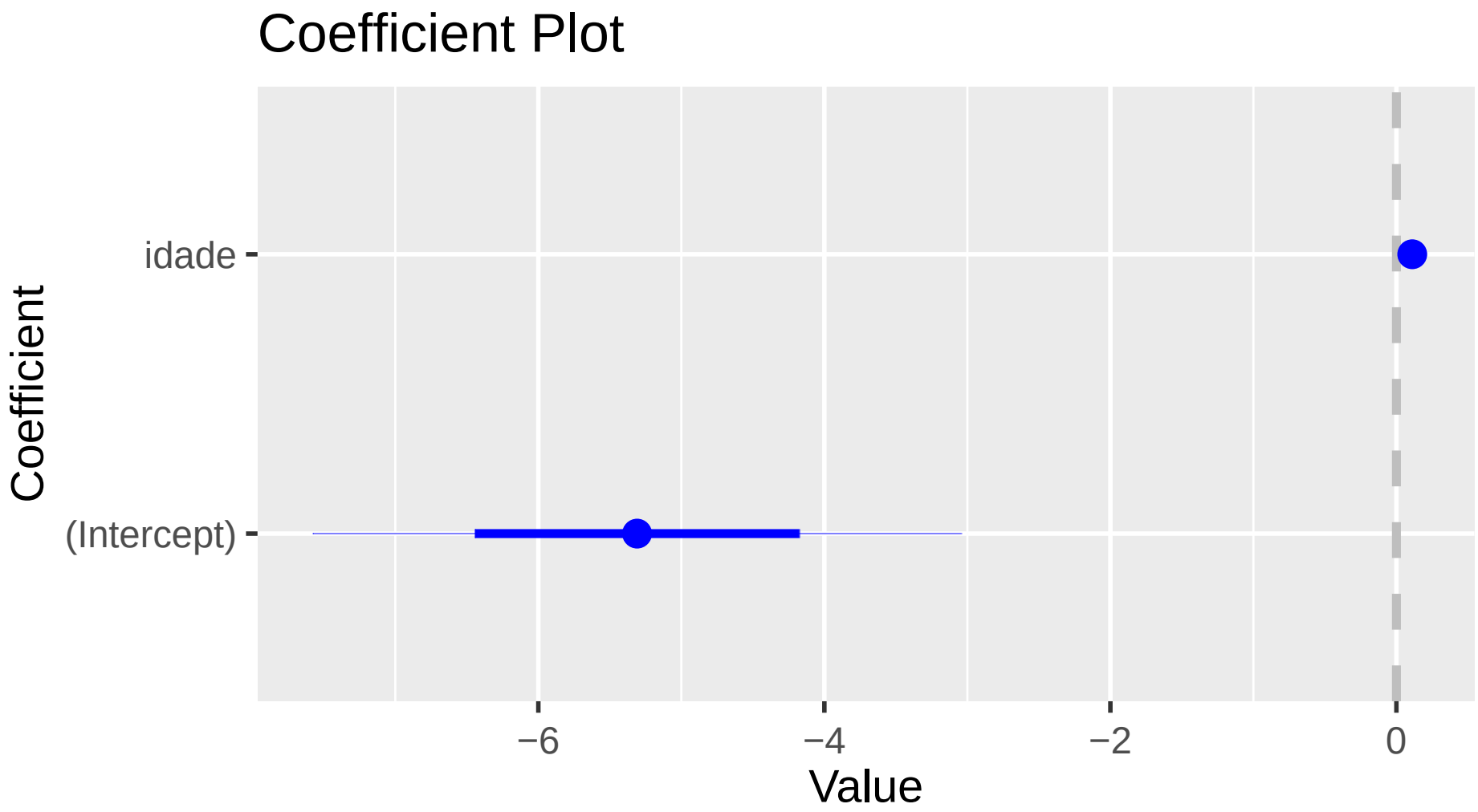
Model Coefficients

```
summary(chdfit1)
```

```
##
## Call:
## glm(formula = chd ~ idade, family = binomial(link = "logit"),
##     data = chdage)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9718  -0.8456  -0.4576   0.8253   2.2859
##
## Coefficients:
##             Estimate Std. Error z value   Pr(>|z|)
## (Intercept) -5.30945    1.13365  -4.683 0.00000282 ***
## idade        0.11092    0.02406   4.610 0.00000402 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 136.66  on 99  degrees of freedom
## Residual deviance: 107.35  on 98  degrees of freedom
## AIC: 111.35
##
## Number of Fisher Scoring iterations: 4
```

Coefficients Plot

```
coefplot::coefplot(chdfit1)
```



Understanding the Coefficients

- Similar to `summary()` of linear regression
- Coefficients themselves represent the log odds that the result would be $Y = 1$.
- You can see on the plot which are positive and which negative
- Graph also indicates the size of the standard error for each independent variable
- To understand the coefficients better, need to calculate the **inverse logit**
- This puts the coefficients in the interval between 0 and 1
  - that is, probability

Inverse Logit

```
invlogit <- function(x) {
   1/(1 + exp(-x))
}
invlogit(chdfit1$coefficients[2])
```

```
##     idade
## 0.5277019
```

- With transformation, we can interpret the results as probabilities
- With a probability $>$ 50%, we can say that age does have a positive relationship with CHD

Deviance and AIC

- 2nd part of the results are equivalent to R^2
  - Measures of quality of the model
- Instead of variance, we use the term *deviance* with `glm()`
- We want to minimize the *residual deviance*
- AIC = Akaike's Information Criterion (here = 111.3530927)
- AIC useful for comparing models
  - Lower number better

This Model

- Residual Deviance = 107.3530927
- AIC = 111.3530927

Second Model for Comparison

- Model with age as a categorical variable – age groups
- Aim is to understand better the probabilities related to age groups than numerical age
  - Are the elderly more likely to have CHD?
- Use `car::recode()`

Age Groups

```
chdage$idgrp <- car::Recode(chdage$idade, "20:29 = '20-29'; 30:34 = '30-34';
                            35:39 = '35-39'; 40:44 = '40-44'; 45:49 = '45-49';
                            50:54 = '50-54'; 55:59 = '55-59'; 60:69 = '60-69'",
                            as.factor = TRUE)
```

Age Group Model

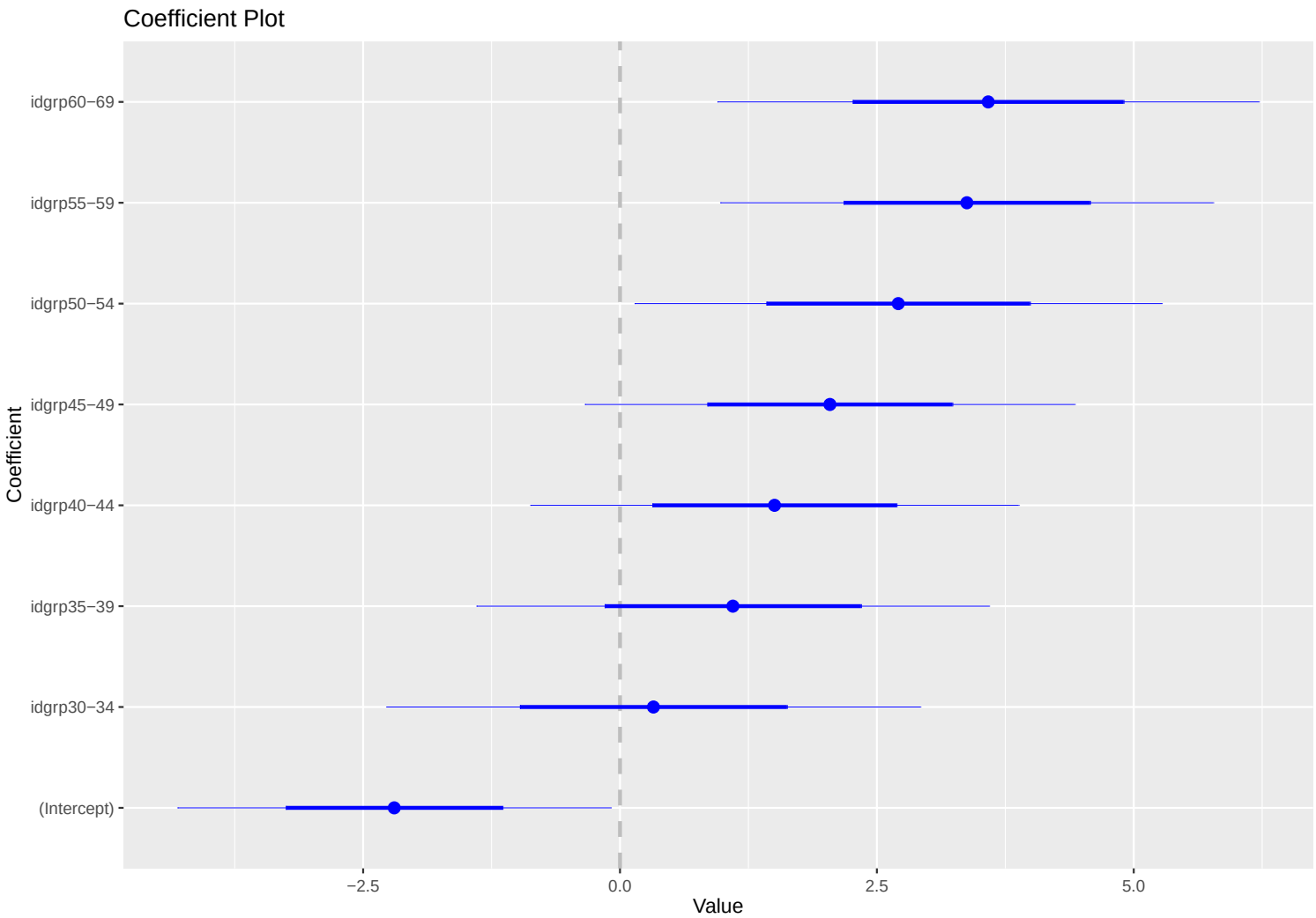
```
chdfit2 <- glm(chd ~  idgrp, data = chdage,
               family = binomial(link = "logit"))
```

Resultados

```
summary(chdfit2)
```

```
##
## Call:
## glm(formula = chd ~ idgrp, family = binomial(link = "logit"),
##     data = chdage)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7941  -0.9005  -0.4590   0.7325   2.1460
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.1972     1.0540  -2.085  0.03710 *
## idgrp30-34    0.3254     1.2992   0.250  0.80221
## idgrp35-39    1.0986     1.2471   0.881  0.37837
## idgrp40-44    1.5041     1.1878   1.266  0.20543
## idgrp45-49    2.0431     1.1918   1.714  0.08649 .
## idgrp50-54    2.7081     1.2823   2.112  0.03470 *
## idgrp55-59    3.3759     1.1991   2.815  0.00487 **
## idgrp60-69    3.5835     1.3175   2.720  0.00653 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 136.66  on 99  degrees of freedom
## Residual deviance: 107.96  on 92  degrees of freedom
## AIC: 123.96
##
## Number of Fisher Scoring iterations: 4
```

Model Coefficients Plot



Elderly Have High Probability of CHD

```
invlogit(coef(chdfit2)[5:8])
```

```
## idgrp45-49 idgrp50-54 idgrp55-59 idgrp60-69
##  0.8852459  0.9375000  0.9669421  0.9729730
```

Which Model Is Better?

- Model 1 – Numeric age
  - Residual Deviance = 107.3530927
  - AIC = 111.3530927
- Model 2 – Categorical Age
  - Residual Deviance = 107.9614654
  - AIC = 123.9614654
- AIC better in the numeric model
- But, the categorical model gives more information about the age groups of interest

Section 2

Example with Multiple Independent Variables

Another CHD Study

- Researchers want to identify factors that cause CHD
- Idendependent Covariates
  - id (Case ID number)
  - age (in years)
  - bmi (body mass index in kg/m^2)
  - gender ($0 =$ male, $1 =$ female)
- 65 cases
- Data - `riscochd.RData`

Load riscochd.RData with load() Function

```
load(here::here("riscochd.RData"))
riscochd <-  riscochd %>%
  mutate(chd = fct_recode(factor(chd), negativo = "0", positivo = "1"),
         genero = fct_recode(factor(genero), masculino = "0", feminino = "1"))
glimpse(riscochd)
```

```
## Rows: 65
## Columns: 5
## $ id     <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 1...
## $ idade  <int> 75, 98, 91, 88, 56, 86, 93, 74, 56, 95, 64, 99, 68, 66, 95, ...
## $ bmi    <dbl> 36.38134, 27.65790, 26.47878, 35.70601, 33.71147, 32.12082, ...
## $ genero <fct> masculino, feminino, feminino, masculino, feminino, masculin...
## $ chd    <fct> positivo, positivo, positivo, positivo, negativo, positivo, ...
```

Exploratory Analysis

```
riscochd %>%
  select(idade, bmi) %>%
  descr(transpose = TRUE,
        stats = c("mean", "sd", "min", "q1", "med", "q3",
                  "max", "iqr", "cv"))
```

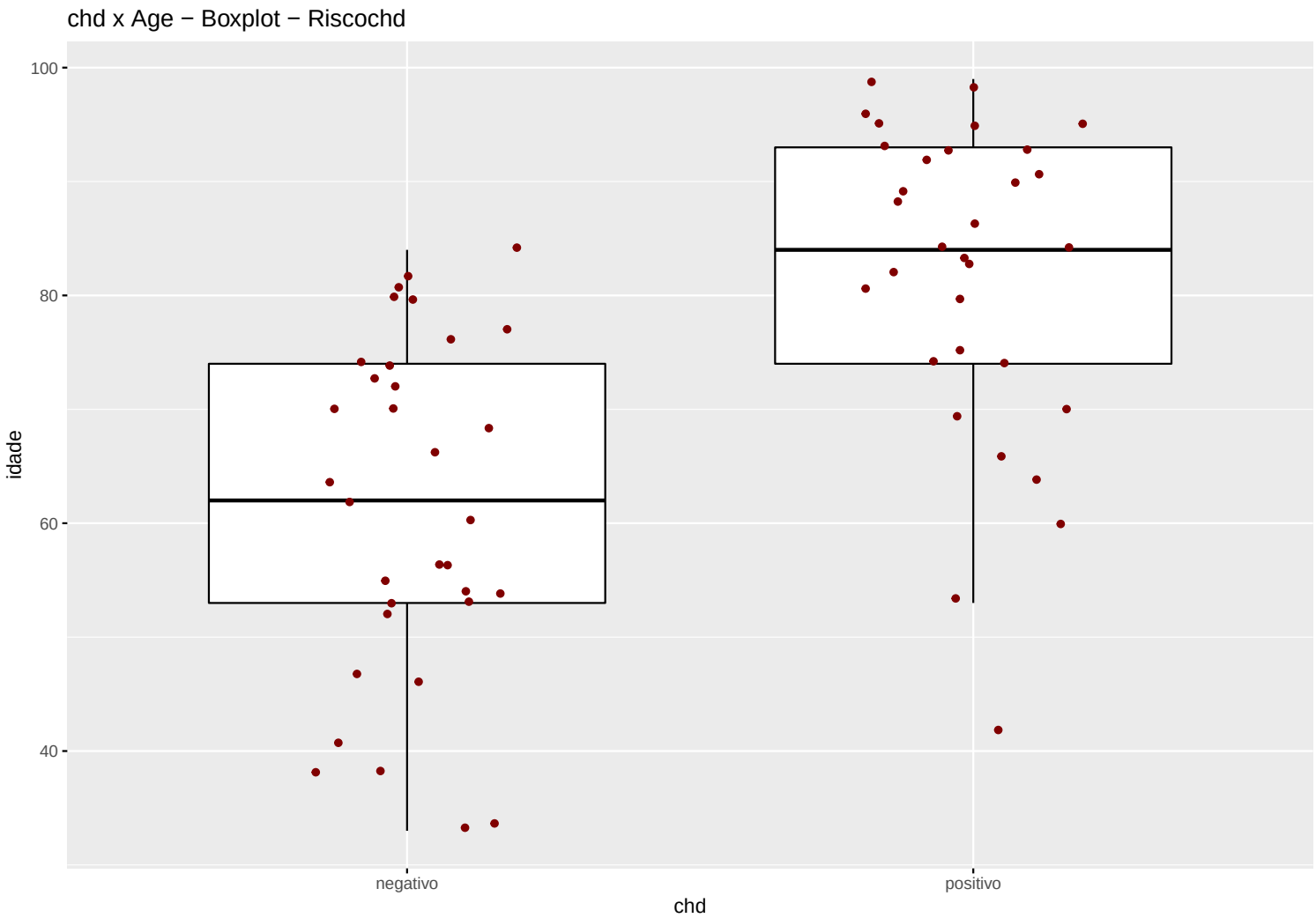
```
## Descriptive Statistics
## riscochd
## N: 65
##
##                Mean    Std.Dev      Min       Q1    Median       Q3      Max      IQR      CV
## ----------- ------- --------- ------- ------- --------- ------- ------- ------- ------
##         bmi   28.42      5.36   16.78   25.18     28.06   31.47   44.94    6.30   0.19
##       idade   71.38     17.67   33.00   56.00     74.00   84.00   99.00   28.00   0.25
```

Categorical Variables

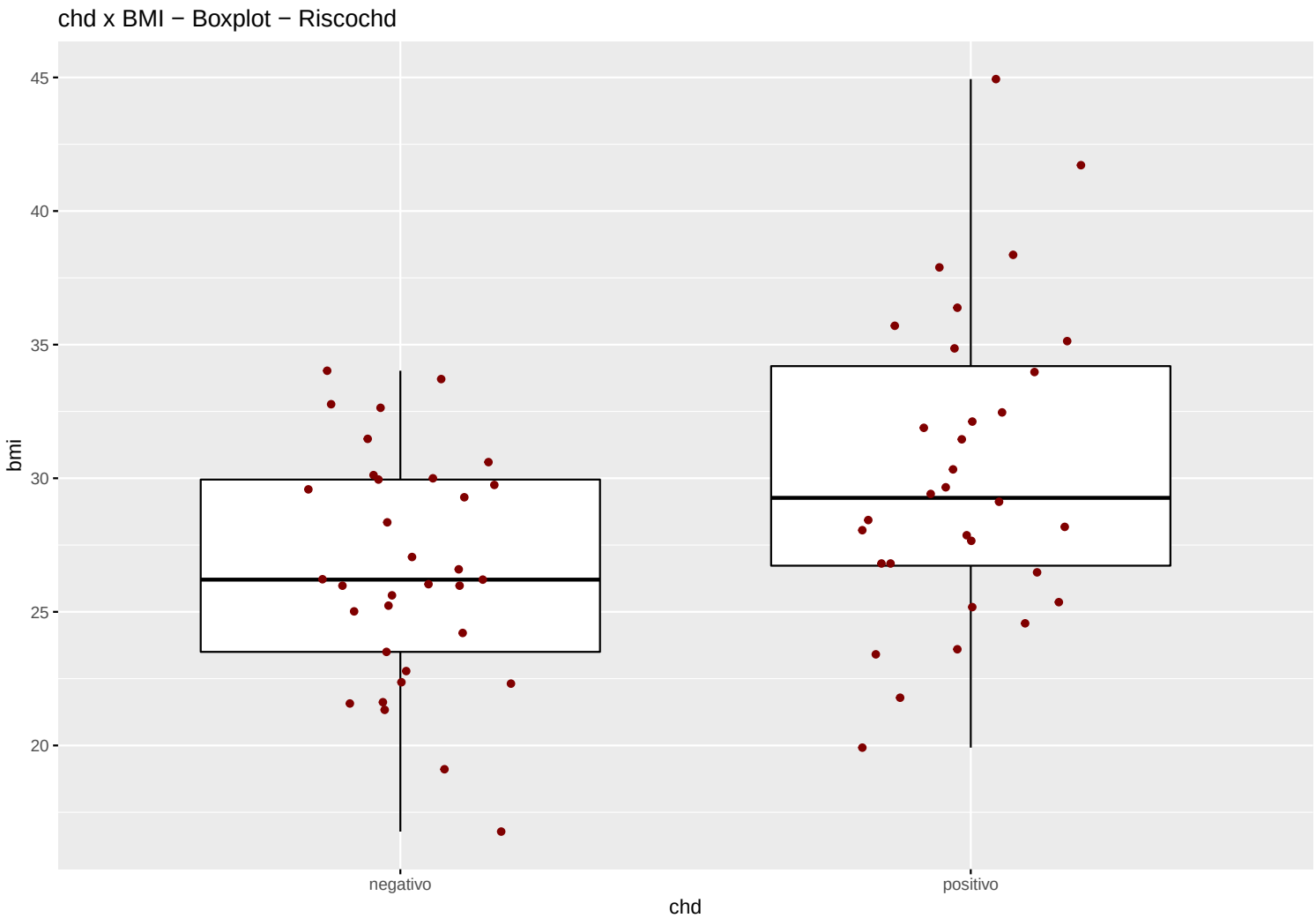
```
riscochd %>%
  select(genero, chd) %>%
  freq()
```

```
## Frequencies
## riscochd$genero
## Type: Factor
##
##                  Freq   % Valid   % Valid Cum.   % Total   % Total Cum.
## --------------- ------ --------- -------------- --------- --------------
##       masculino     41     63.08          63.08     63.08          63.08
##        feminino     24     36.92         100.00     36.92         100.00
##            <NA>      0                               0.00         100.00
##           Total     65    100.00         100.00    100.00         100.00
##
## riscochd$chd
## Type: Factor
##
##                  Freq   % Valid   % Valid Cum.   % Total   % Total Cum.
## --------------- ------ --------- -------------- --------- --------------
##       negativo     33     50.77          50.77     50.77          50.77
##       positivo     32     49.23         100.00     49.23         100.00
##            <NA>      0                               0.00         100.00
##           Total     65    100.00         100.00    100.00         100.00
```

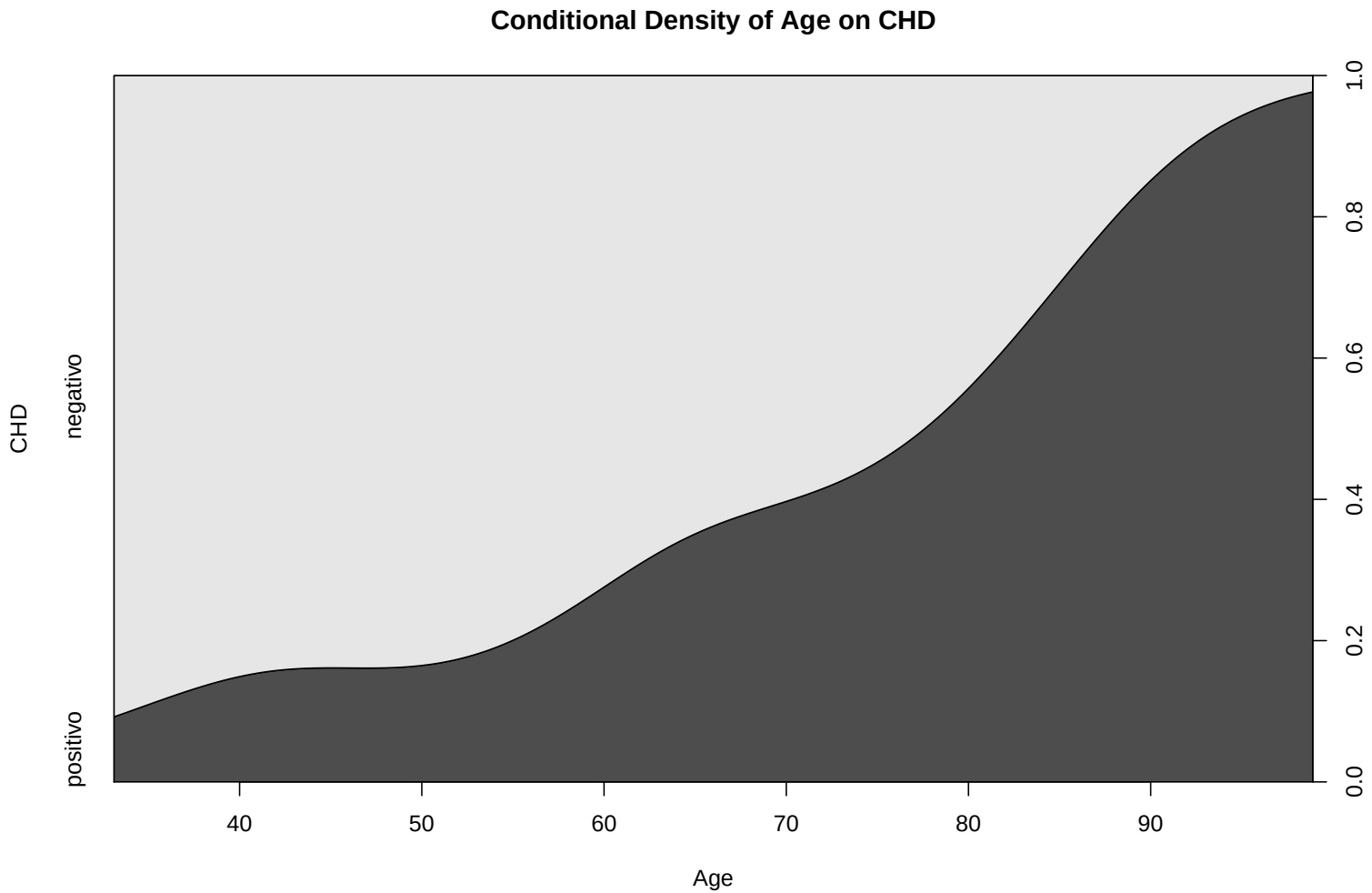
Boxplot of Age

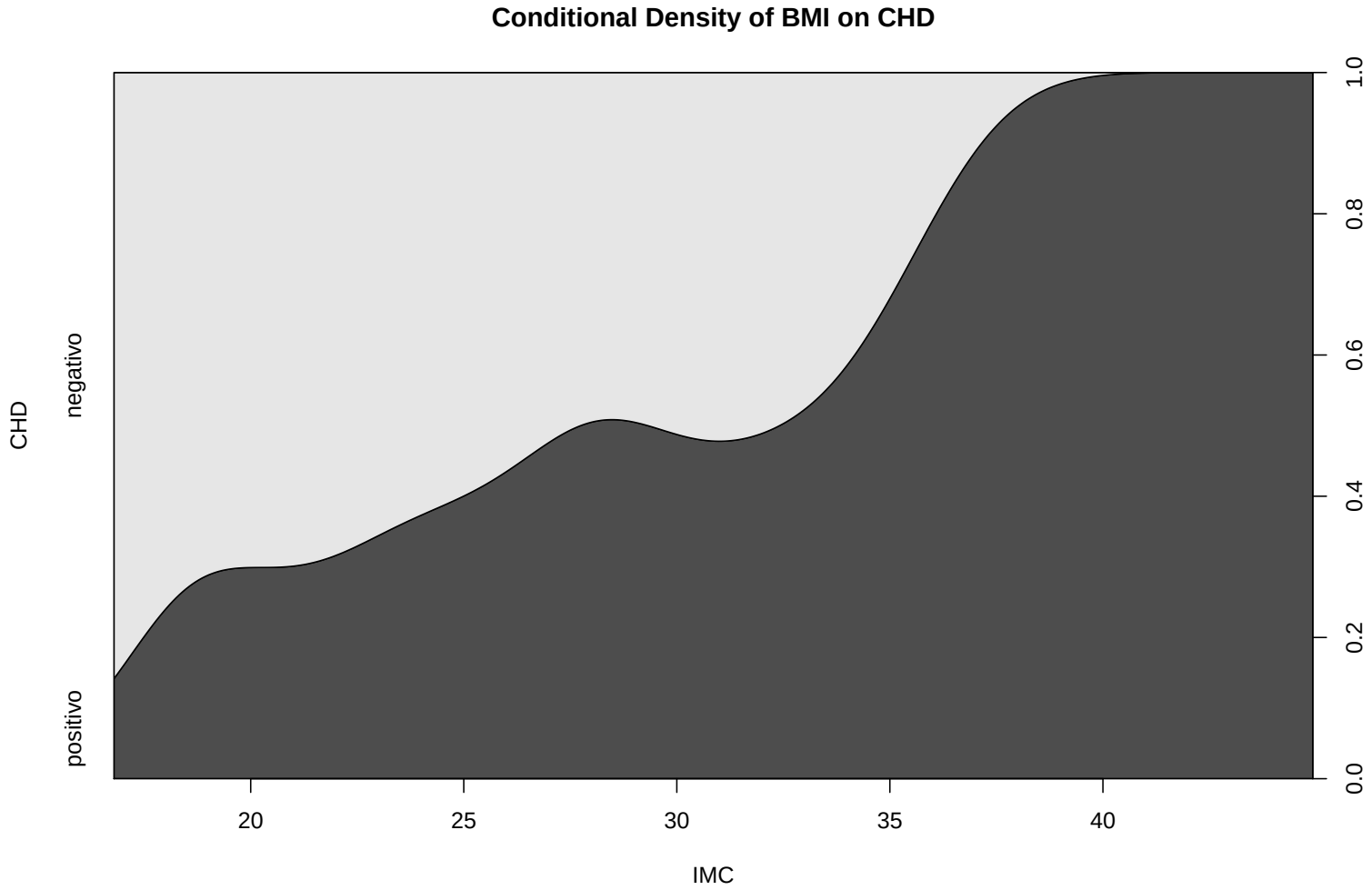


Boxplot of BMI



Conditional Density Plot – Age




Conditional Density of BMI on CHD
CHD
negativo
positivo
0.0
0.2
0.4
0.6
0.8
1.0
20
25
30
35
40
IMC

Model 1 – All the Independent Variables

```
chdfit3 <- glm(chd ~ idade + bmi + genero, data = riscochd,
               family = binomial(link = "logit"))
summary(chdfit3)
```

```
##
## Call:
## glm(formula = chd ~ idade + bmi + genero, family = binomial(link = "logit"),
##     data = riscochd)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.84596  -0.48371  -0.05345   0.48149   2.46001
##
## Coefficients:
##                 Estimate Std. Error z value  Pr(>|z|)
## (Intercept)    -20.64336    5.06903  -4.072 0.0000465 ***
## idade            0.14814    0.03822   3.876  0.000106 ***
## bmi              0.34613    0.10189   3.397  0.000681 ***
## generofeminino   0.45202    0.77568   0.583  0.560069
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 90.094  on 64  degrees of freedom
## Residual deviance: 43.886  on 61  degrees of freedom
## AIC: 51.886
##
## Number of Fisher Scoring iterations: 6
```

Model 2 – Using Only the Age Variable

```r
chdfit4 <- glm(chd ~ idade, data = riscochd,
               family = binomial(link = "logit"))
summary(chdfit4)
```

```
##
## Call:
## glm(formula = chd ~ idade, family = binomial(link = "logit"),
##     data = riscochd)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6471  -0.7813  -0.2121   0.7718   2.4418
##
## Coefficients:
##             Estimate Std. Error z value  Pr(>|z|)
## (Intercept) -6.91677    1.79219  -3.859  0.000114 ***
## idade        0.09495    0.02393   3.968 0.0000725 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 90.094  on 64  degrees of freedom
## Residual deviance: 64.000  on 63  degrees of freedom
## AIC: 68
##
## Number of Fisher Scoring iterations: 5
```

Second Model Compared to the First

- AIC increased in the age only model
- Model had lower quality

Model 3 – Using the Age and BMI Variables

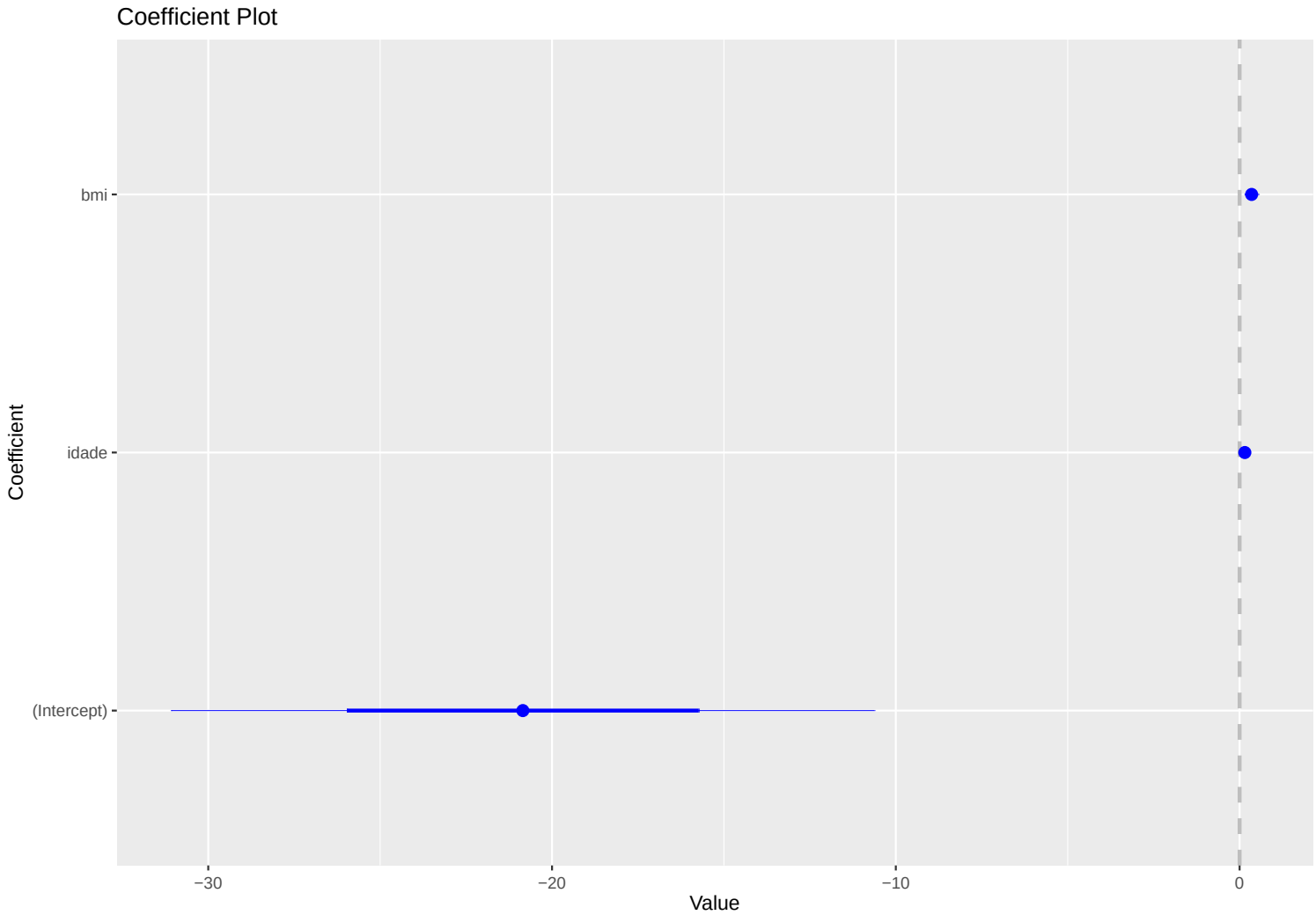
```r
chdfit5 <- glm(chd ~ idade + bmi, data = riscochd,
               family = binomial(link = "logit"))
summary(chdfit5)
```

```
##
## Call:
## glm(formula = chd ~ idade + bmi, family = binomial(link = "logit"),
##     data = riscochd)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.94448  -0.51392  -0.05453   0.52326   2.40266
##
## Coefficients:
##              Estimate Std. Error z value  Pr(>|z|)
## (Intercept) -20.84877    5.11434  -4.077 0.0000457 ***
## idade         0.15229    0.03819   3.988 0.0000667 ***
## bmi           0.35020    0.10196   3.435  0.000593 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 90.094  on 64  degrees of freedom
## Residual deviance: 44.225  on 62  degrees of freedom
## AIC: 50.225
##
## Number of Fisher Scoring iterations: 6
```

New Model Performance

- Of all three models, best AIC (50.2246163)
- Residual Deviance very close to (but a bit higher) than the first model

Plot of the Coefficients of Final Model



Results Translated to Probability and Odds

```
paste("Odds Ratio:", round(exp(coef(chdfit5)), 3)) # Calculate the odds
```

```
## [1] "Odds Ratio: 0"     "Odds Ratio: 1.165" "Odds Ratio: 1.419"
```

```
exp(confint(chdfit5))
```

```
##                    2.5 %         97.5 %
## (Intercept) 6.449713e-15 0.000004578083
## idade       1.092511e+00 1.272408489774
## bmi         1.192024e+00 1.794849303037
```

```
paste("Probability of Occurrence:", round(invlogit(chdfit5$coefficients),3))
```

```
## [1] "Probability of Occurrence: 0"     "Probability of Occurrence: 0.538"
## [3] "Probability of Occurrence: 0.587"
```

Conclusion about `riscochd`

- The two variables in the last model have more than 50% probability of being risks for CHD
- Logistic regression models are difficult to interpret
  - Log Odds, Odds ratios, AIC, etc.
- Logistic regression important technique that you will see frequently

Section 3

Third Example of Logistic Regression

Breast Cancer Diagnosis Model

- Data come from a Wisconsin study on breast cancer
- Characteristics of breast cancer tumors
- Dependent variable: diagnosis (`diag`)
- Model more realistic than earlier
  - More covariates
  - Presence of NA's

Covariates – Tumor Characteristics

- Come from analysis of images based on fine needle aspiration
- Characteristics
  - Sample ID (code number)
  - Clump thickness
  - Uniformity of cell size
  - Uniformity of cell shape
  - Marginal adhesion
  - Single epithelial cell size
  - Number of bare nuclei
  - Bland chromatin
  - Number of normal nuclei
  - Mitosis

Load Data

```
bc_data <- read.table(here::here("breast-cancer-wisconsin-data.txt"),
                      header = FALSE,
                      sep = ",",
                      na.strings = "?")
colnames(bc_data) <- c("sample_code_number",
                       "clump_thickness",
                       "uniformity_of_cell_size",
                       "uniformity_of_cell_shape",
                       "marginal_adhesion",
                       "single_epithelial_cell_size",
                       "bare_nuclei",
                       "bland_chromatin",
                       "normal_nucleoli",
                       "mitosis",
                       "diag")


bc_data$diag <- ifelse(bc_data$diag == "2", "benign",
                       ifelse(bc_data$diag == "4", "malignant", NA))
```

Data

```
glimpse(bc_data)
```

```
## Rows: 699
## Columns: 11
## $ sample_code_number          <int> 1000025, 1002945, 1015425, 1016277, 101...
## $ clump_thickness             <int> 5, 5, 3, 6, 4, 8, 1, 2, 2, 4, 1, 2, 5, ...
## $ uniformity_of_cell_size     <int> 1, 4, 1, 8, 1, 10, 1, 1, 1, 2, 1, 1, 3,...
## $ uniformity_of_cell_shape    <int> 1, 4, 1, 8, 1, 10, 1, 2, 1, 1, 1, 1, 3,...
## $ marginal_adhesion           <int> 1, 5, 1, 1, 3, 8, 1, 1, 1, 1, 1, 1, 3, ...
## $ single_epithelial_cell_size <int> 2, 7, 2, 3, 2, 7, 2, 2, 2, 2, 1, 2, 2, ...
## $ bare_nuclei                 <int> 1, 10, 2, 4, 1, 10, 10, 1, 1, 1, 1, 1, ...
## $ bland_chromatin             <int> 3, 3, 3, 3, 3, 9, 3, 3, 1, 2, 3, 2, 4, ...
## $ normal_nucleoli             <int> 1, 2, 1, 7, 1, 7, 1, 1, 1, 1, 1, 1, 4, ...
## $ mitosis                     <int> 1, 1, 1, 1, 1, 1, 1, 1, 5, 1, 1, 1, 1, ...
## $ diag                        <chr> "benign", "benign", "benign", "benign",...
```

Analysis of NAs – What Will We Do with Them

- How many NAs are in the data?

```
sum(is.na(bc_data))
```

```
## [1] 16
```

- Are all of them in the `bare_nuclei` variable?

How Many Cases Do We Lose If We Take Out the NAs?

```
glue::glue("Número de casos perdidos: ", nrow(bc_data[is.na(bc_data), ]))
```

```
## Número de casos perdidos: 16
```

```
glue::glue("Tamanho da base final: ", dim(drop_na(bc_data))[1])
```

```
## Tamanho da base final: 683
```

Options to Resolve NAs

- Eliminate cases with NA - `tidyr::drop_na()`
- Fill in NAs com neighboring values - `tidyr::fill()`
- Fill in with another value - `tidyr::replace_na()`
  - Value that you decide
  - Eg. 0 (`x <- x %>% mutate_all(replace_na, 0)`)
- Impute values with `mice` package

Impute Values with `mice::mice`

- Multivariate Imputation by Chained Equations
- Create imputed data for incomplete multivariate data
  - Gibbs Sampling (Bayesian technique)
  - Generates plausible synthetic values based on other variables in the data set
- Imputation introduces more uncertainty in the model

```r
descr(bc_data$bare_nuclei, transpose = TRUE, # todos NA vem de bare_nuclei
      stats = c("mean", "sd", "med", "min", "max", "n.valid"))
```

```
## Descriptive Statistics
## bc_data$bare_nuclei
## N: 699
##
##                      Mean   Std.Dev   Median    Min     Max   N.Valid
## ----------------- ------ --------- -------- ------ ------- ---------
##       bare_nuclei   3.54      3.64     1.00   1.00   10.00    683.00
```

```r
a_numero <- function(x) as.numeric(as.character(x))
mod_cols <- colnames(bc_data[2:10])
bc_data <- bc_data %>%
  mutate_at(mod_cols, ~a_numero(.), na.rm = TRUE)
dataset_impute <- mice::mice(bc_data[, 2:10],  print = FALSE)
bc_data <- cbind(diag = bc_data$diag, mice::complete(dataset_impute, 1))
descr(bc_data$bare_nuclei, transpose = TRUE, # todos NA vem de bare_nuclei
      stats = c("mean", "sd", "med", "min", "max", "n.valid"))
```

```
## Descriptive Statistics
## bc_data$bare_nuclei
## N: 699
##
##                      Mean   Std.Dev   Median    Min     Max   N.Valid
## ----------------- ------ --------- -------- ------ ------- ---------
##       bare_nuclei   3.51      3.62     1.00   1.00   10.00    699.00
```

Summary of Diagnoses

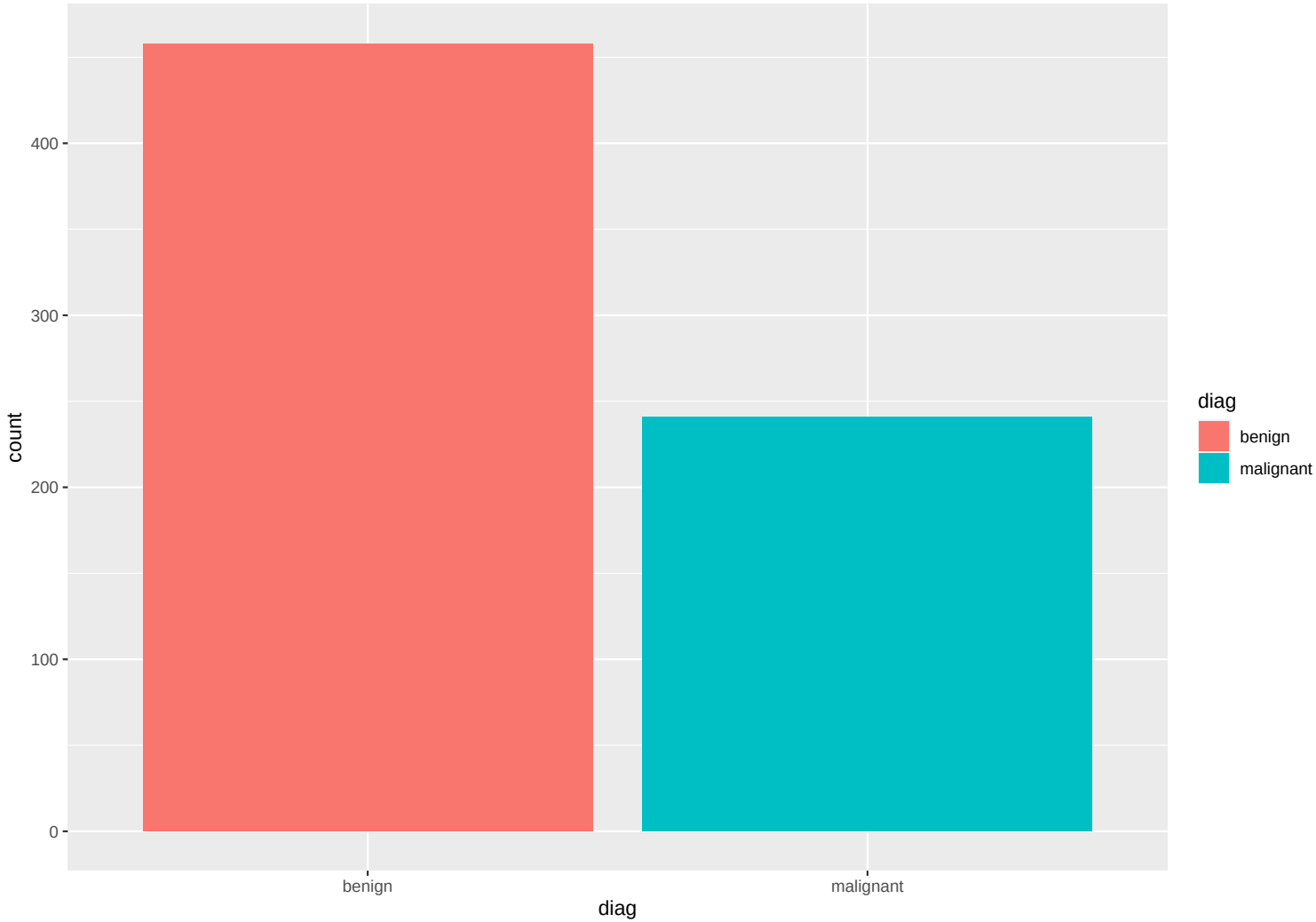
- Convert `diag` to a `factor`
- How many benign and malignant cases are there?

```
bc_data$diag <- as.factor(bc_data$diag)
 summary(bc_data$diag)
```

```
##    benign malignant
##       458       241
```

Plot of Diagnoses

```
brgr1 <- ggplot(bc_data, aes(x = diag, fill = diag)) + geom_bar()
brgr1
```



Unequal `diag` Classes

- Normally need an adjustment to deal with inequality
- But, not today

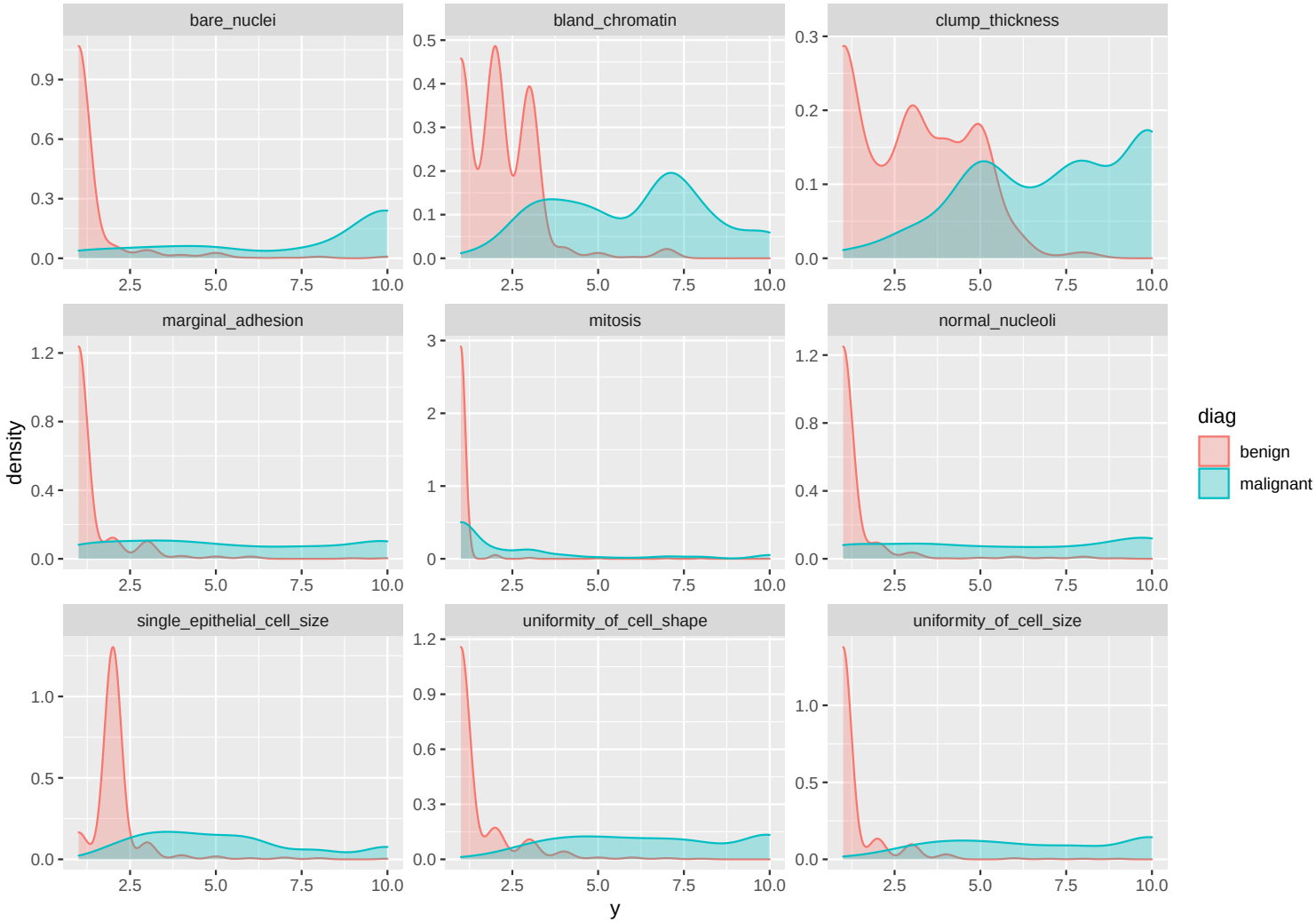
Exploration of Some of the Covariates

```
bc_data %>%
  select(clump_thickness:mitosis) %>%
  descr(transpose = TRUE,
        stats = c("mean", "sd", "min", "q1", "med", "q3",
                  "max", "iqr", "cv"))
```

```
## Descriptive Statistics
## bc_data
## N: 699
##
##                                 Mean   Std.Dev    Min     Q1   Median     Q3    Max    IQR     CV
## -------------------------------- ------ --------- ------ ------ -------- ------ ------- ------ ------
##                      bare_nuclei   3.51      3.62   1.00   1.00     1.00   6.00   10.00   5.00   1.03
##                  bland_chromatin   3.44      2.44   1.00   2.00     3.00   5.00   10.00   3.00   0.71
##                  clump_thickness   4.42      2.82   1.00   2.00     4.00   6.00   10.00   4.00   0.64
##                marginal_adhesion   2.81      2.86   1.00   1.00     1.00   4.00   10.00   3.00   1.02
##                          mitosis   1.59      1.72   1.00   1.00     1.00   1.00   10.00   0.00   1.08
##                 normal_nucleoli   2.87      3.05   1.00   1.00     1.00   4.00   10.00   3.00   1.07
##       single_epithelial_cell_size   3.22      2.21   1.00   2.00     2.00   4.00   10.00   2.00   0.69
##          uniformity_of_cell_shape   3.21      2.97   1.00   1.00     1.00   5.00   10.00   4.00   0.93
##           uniformity_of_cell_size   3.13      3.05   1.00   1.00     1.00   5.00   10.00   4.00   0.97
```

Plot of Covariates with the Diagnosis das Covariáveis com a Diagnose

```
gr_covars <- gather(bc_data, x, y, clump_thickness:mitosis) %>%
  ggplot(aes(x = y, color = diag, fill = diag)) +
    geom_density(alpha = 0.3) +
    facet_wrap( ~ x, scales = "free", ncol = 3)
```


bare_nuclei
bland_chromatin
clump_thickness
marginal_adhesion
mitosis
normal_nucleoli
single_epithelial_cell_size
uniformity_of_cell_shape
uniformity_of_cell_size
density
y
diag
benign
malignant

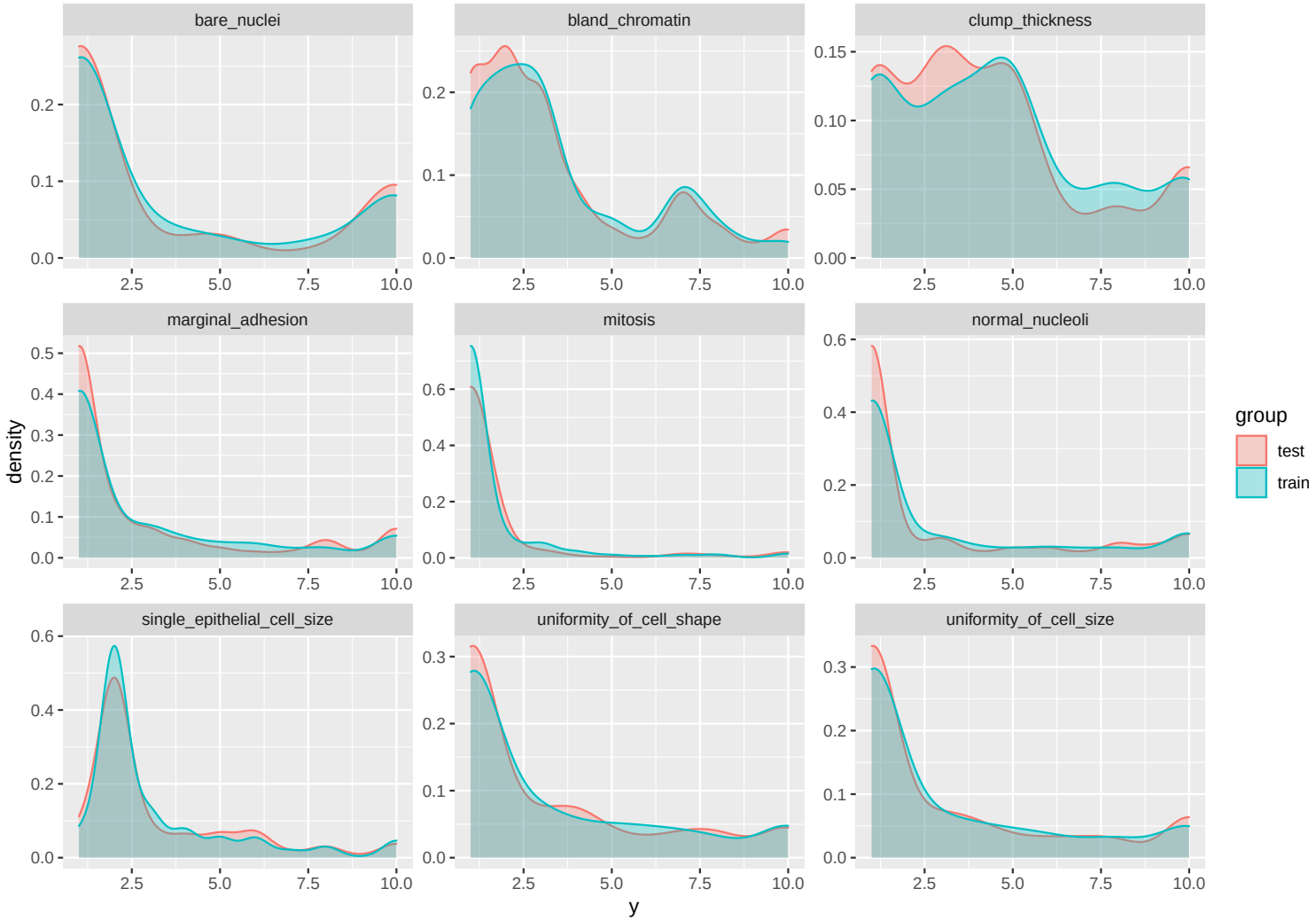
Build Model with `caret`

- Funções para apoiar machine learning
- Pode conduzir todo a análise dentro de `caret`
- No grupos dos pacotes iniciais

Create Training and Test Data

```
set.seed(42)
index <- caret::createDataPartition(bc_data$diag, p = 0.7, list = FALSE)
train_data <- bc_data[index, ]
test_data <- bc_data[-index, ]
```

Do the Training and Test Sets Reflect the Same Data?



Train Control – Cross Validation

- Before training our model, need to decide what type of validation we want to use
  - bootstrap, k-fold cross validation
- We will use *10-fold cross validation*
- Will strengthen the validation process by repeating it 10 times

trainControl()

```
set.seed(42)
control <- trainControl(method = "repeatedcv",
                        number = 10,
                        repeats = 10,
                        savePredictions = TRUE,
                        verboseIter = FALSE)
```

Train the Model with Logistic Regression

```
model_glm <- caret::train(diag ~ .,
                          data = train_data,
                          method = "glm",
                          preProcess = c("scale", "center"),
                          trControl = control)
```

Model

```
model_glm
```

```
## Generalized Linear Model
##
## 490 samples
##   9 predictor
##   2 classes: 'benign', 'malignant'
##
## Pre-processing: scaled (9), centered (9)
## Resampling: Cross-Validated (10 fold, repeated 10 times)
## Summary of sample sizes: 441, 441, 441, 441, 441, 441, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.9538864  0.8975163
```

Summary of Model Results

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.2699  -0.1647  -0.0840   0.0415   2.4068
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -1.15008    0.30601  -3.758 0.000171 ***
## clump_thickness              1.45679    0.40877   3.564 0.000366 ***
## uniformity_of_cell_size     -0.37247    0.63538  -0.586 0.557737
## uniformity_of_cell_shape     1.32760    0.71892   1.847 0.064798 .
## marginal_adhesion            0.79412    0.34782   2.283 0.022424 *
## single_epithelial_cell_size -0.06482    0.35409  -0.183 0.854761
## bare_nuclei                  1.05272    0.34924   3.014 0.002576 **
## bland_chromatin              1.23724    0.42776   2.892 0.003823 **
## normal_nucleoli              0.24995    0.35824   0.698 0.485361
## mitosis                      0.99718    0.48203   2.069 0.038571 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 631.35  on 489  degrees of freedom
## Residual deviance: 100.96  on 480  degrees of freedom
## AIC: 120.96
##
## Number of Fisher Scoring iterations: 8
```

Can the Model Predict the Results We Already Know?

- `predict()` function
  - Using the model and values we can use for prediction
- First, applied ot the `train` set as an example
- More interesting – `test` set
  - Because the model has never seen these data
- **Acid Test**

Predictions

```
predtr <- predict(model_glm, train_data)
predtest <- predict(model_glm, test_data)
tabyl(predtest) %>% adorn_pct_formatting()
```

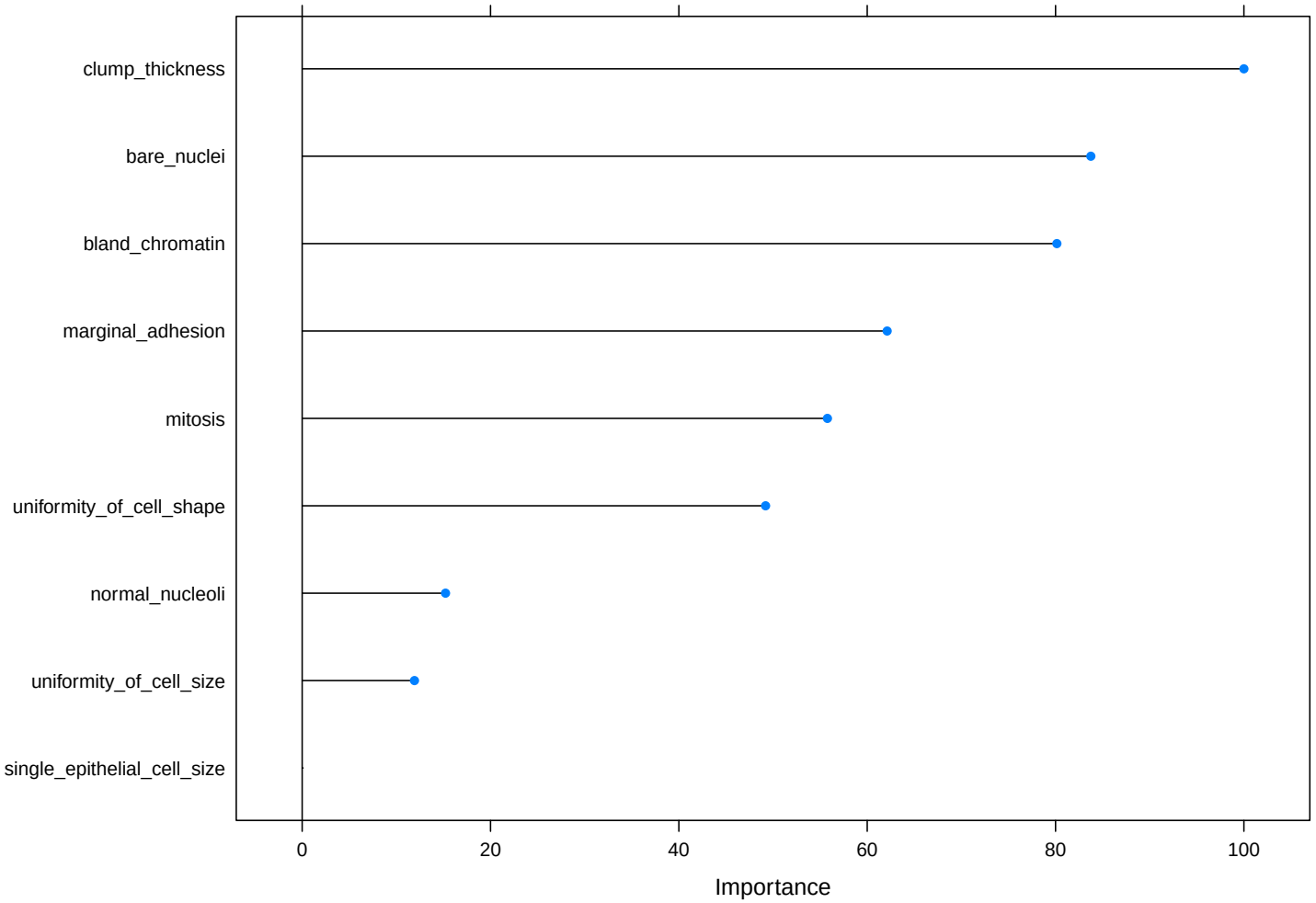
```
##   predtest   n percent
##     benign 139   66.5%
##  malignant  70   33.5%
```

```
tabyl(predtr) %>% adorn_pct_formatting()
```

```
##     predtr   n percent
##     benign 322   65.7%
##  malignant 168   34.3%
```

Which Variables Are Important in the Model?

```
plot(caret::varImp(model_glm))
```



Confusion Matrix - A Truth Table

		True condition	
	Total population	Condition positive	Condition negative
Predicted condition	Predicted condition positive	**True positive**	**False positive,** Type I error
	Predicted condition negative	**False negative,** Type II error	**True negative**

- Way of comparing predictions to the truth
- If the predictions are not correct, they either suffer from Type I or Type II errors
  - Type I - False positive
  - Type II - False negative

Calculations You Can Do with the Confusion Matrix

		True condition			
	Total population	Condition positive	Condition negative	Prevalence = $\frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$
Predicted condition	Predicted condition positive	**True positive**	**False positive**, Type I error	Positive predictive value (PPV), Precision = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Predicted condition positive}}$
	Predicted condition negative	**False negative**, Type II error	**True negative**	False omission rate (FOR) = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Predicted condition negative}}$	Negative predictive value (NPV) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Predicted condition negative}}$
		True positive rate (TPR), Recall, Sensitivity, probability of detection, Power = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	Positive likelihood ratio (LR+) = $\frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) = $\frac{\text{LR+}}{\text{LR−}}$
		False negative rate (FNR), Miss rate = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	Specificity (SPC), Selectivity, True negative rate (TNR) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$	Negative likelihood ratio (LR−) = $\frac{\text{FNR}}{\text{TNR}}$	F_1 score = $2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$

Predictions Based on the Test Set – Confusion Matrix

```
confusionMatrix(predtest, test_data$diag, positive = "malignant")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  benign malignant
##   benign       135         4
##   malignant      2        68
##
##                Accuracy : 0.9713
##                  95% CI : (0.9386, 0.9894)
##     No Information Rate : 0.6555
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.936
##
##  Mcnemar's Test P-Value : 0.6831
##
##             Sensitivity : 0.9444
##             Specificity : 0.9854
##          Pos Pred Value : 0.9714
##          Neg Pred Value : 0.9712
##              Prevalence : 0.3445
##          Detection Rate : 0.3254
##    Detection Prevalence : 0.3349
##       Balanced Accuracy : 0.9649
##
##        'Positive' Class : malignant
##
```

Predictions Based on the Training Set – Confusion Matrix

```r
confusionMatrix(predtr, train_data$diag, positive = "malignant")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  benign malignant
##   benign       312        10
##   malignant      9       159
##
##                Accuracy : 0.9612
##                  95% CI : (0.9401, 0.9765)
##     No Information Rate : 0.6551
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.9141
##
##  Mcnemar's Test P-Value : 1
##
##             Sensitivity : 0.9408
##             Specificity : 0.9720
##          Pos Pred Value : 0.9464
##          Neg Pred Value : 0.9689
##              Prevalence : 0.3449
##          Detection Rate : 0.3245
##    Detection Prevalence : 0.3429
##       Balanced Accuracy : 0.9564
##
##        'Positive' Class : malignant
##
```

Section 4

Next Week

Next Week

- ROC Curves
- Other Classification Algoritms
- Principal Components Analysis
- Cluster Analysis