Predictive Bank Client Deposit Rate Analysis

Mykolas Motiejūnas

2025-04-26

Packages

```r
library(readr)
library(dplyr)
library(DescTools)
library(gmodels)
library(ggplot2)
library(GGally)
library(gridExtra)
library(tidyr)
library(bestNormalize)
library(caret)
library(pROC)
```

Predefined functions

```r
create_bar_plot <- function(data, var_name) {
  freq_table <- data %>%
    group_by(!!sym(var_name), subscribed) %>%
    summarise(count = n(), .groups = "drop") %>%
    group_by(!!sym(var_name)) %>%
    mutate(prop = count / sum(count))

  p <- ggplot(freq_table, aes(x = !!sym(var_name), y = prop, fill = subscribed)) +
    geom_bar(stat = "identity", position = "fill") +
    scale_fill_manual(values = c("TRUE" = "#4CAF50", "FALSE" = "#F44336")) +
    labs(x = var_name,
         y = "",
         fill = "Subscribed") +
    theme_minimal() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1),
          plot.title = element_text(size = 10),
          legend.position = "none")

  return(p)
}
```

Importing data

```r
bank_full <- read_delim("bankData/bank-full.csv",
    delim = ";", escape_double = FALSE, trim_ws = TRUE)
```

Data cleaning

```r
head(bank_full)
```

```
## # A tibble: 6 x 17
##     age job        marital education default balance housing loan  contact   day
##   <dbl> <chr>      <chr>   <chr>     <chr>     <dbl> <chr>   <chr> <chr>   <dbl>
## 1    58 management married tertiary  no         2143 yes     no    unknown     5
## 2    44 technician single  secondary no           29 yes     no    unknown     5
## 3    33 entrepren~ married secondary no            2 yes     yes   unknown     5
## 4    47 blue-coll~ married unknown   no         1506 yes     no    unknown     5
## 5    33 unknown    single  unknown   no            1 no      no    unknown     5
## 6    35 management married tertiary  no          231 yes     no    unknown     5
## # i 7 more variables: month <chr>, duration <dbl>, campaign <dbl>, pdays <dbl>,
## #   previous <dbl>, poutcome <chr>, y <chr>
```

```r
tail(bank_full)
```

```
## # A tibble: 6 x 17
##     age job        marital education default balance housing loan  contact   day
##   <dbl> <chr>      <chr>   <chr>     <chr>     <dbl> <chr>   <chr> <chr>   <dbl>
## 1    25 technician single  secondary no          505 no      yes   cellul~    17
## 2    51 technician married tertiary  no          825 no      no    cellul~    17
## 3    71 retired    divorc~ primary   no         1729 no      no    cellul~    17
## 4    72 retired    married secondary no         5715 no      no    cellul~    17
## 5    57 blue-coll~ married secondary no          668 no      no    teleph~    17
## 6    37 entrepren~ married secondary no         2971 no      no    cellul~    17
## # i 7 more variables: month <chr>, duration <dbl>, campaign <dbl>, pdays <dbl>,
## #   previous <dbl>, poutcome <chr>, y <chr>
```

```r
str(bank_full)
```

```
## spc_tbl_ [45,211 x 17] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ age      : num [1:45211] 58 44 33 47 33 35 28 42 58 43 ...
##  $ job      : chr [1:45211] "management" "technician" "entrepreneur" "blue-collar" ...
##  $ marital  : chr [1:45211] "married" "single" "married" "married" ...
##  $ education: chr [1:45211] "tertiary" "secondary" "secondary" "unknown" ...
##  $ default  : chr [1:45211] "no" "no" "no" "no" ...
##  $ balance  : num [1:45211] 2143 29 2 1506 1 ...
##  $ housing  : chr [1:45211] "yes" "yes" "yes" "yes" ...
##  $ loan     : chr [1:45211] "no" "no" "yes" "no" ...
##  $ contact  : chr [1:45211] "unknown" "unknown" "unknown" "unknown" ...
##  $ day      : num [1:45211] 5 5 5 5 5 5 5 5 5 5 ...
##  $ month    : chr [1:45211] "may" "may" "may" "may" ...
##  $ duration : num [1:45211] 261 151 76 92 198 139 217 380 50 55 ...
##  $ campaign : num [1:45211] 1 1 1 1 1 1 1 1 1 1 ...
##  $ pdays    : num [1:45211] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
##  $ previous : num [1:45211] 0 0 0 0 0 0 0 0 0 0 ...
##  $ poutcome : chr [1:45211] "unknown" "unknown" "unknown" "unknown" ...
##  $ y        : chr [1:45211] "no" "no" "no" "no" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   age = col_double(),
##   ..   job = col_character(),
##   ..   marital = col_character(),
##   ..   education = col_character(),
```

```
##   ..   default = col_character(),
##   ..   balance = col_double(),
##   ..   housing = col_character(),
##   ..   loan = col_character(),
##   ..   contact = col_character(),
##   ..   day = col_double(),
##   ..   month = col_character(),
##   ..   duration = col_double(),
##   ..   campaign = col_double(),
##   ..   pdays = col_double(),
##   ..   previous = col_double(),
##   ..   poutcome = col_character(),
##   ..   y = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

A first look at the data shows us that many of the provided columns have an incorrect data type. For example, default and marital status are set as character data types when they should be factors.

```
bank_full <- bank_full %>%
  mutate(
    job = if_else(job == "admin.", "admin", job),
    across(c(marital, job, education, contact, poutcome, day), as.factor),
    across(c(default, housing, loan, y), ~ .x == "yes"),
    month = factor(month, levels = c("jan", "feb", "mar", "apr", "may", "jun",
                                     "jul", "aug", "sep", "oct", "nov", "dec")),
    job = relevel(job, ref = "unemployed"),
    marital = relevel(marital, ref = "single"),
    education = relevel(education, ref = "unknown"),
    contact = relevel(contact, ref = "unknown"),
    poutcome = relevel(poutcome, ref = "unknown")
  )

str(bank_full)
```

```
## tibble [45,211 x 17] (S3: tbl_df/tbl/data.frame)
##  $ age      : num [1:45211] 58 44 33 47 33 35 28 42 58 43 ...
##  $ job      : Factor w/ 12 levels "unemployed","admin",..: 6 11 4 3 12 6 6 4 7 11 ...
##  $ marital  : Factor w/ 3 levels "single","divorced",..: 3 1 3 3 1 3 1 2 3 1 ...
##  $ education: Factor w/ 4 levels "unknown","primary",..: 4 3 3 1 1 4 4 4 2 3 ...
##  $ default  : logi [1:45211] FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ balance  : num [1:45211] 2143 29 2 1506 1 ...
##  $ housing  : logi [1:45211] TRUE TRUE TRUE TRUE FALSE TRUE ...
##  $ loan     : logi [1:45211] FALSE FALSE TRUE FALSE FALSE FALSE ...
##  $ contact  : Factor w/ 3 levels "unknown","cellular",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ day      : Factor w/ 31 levels "1","2","3","4",..: 5 5 5 5 5 5 5 5 5 5 ...
##  $ month    : Factor w/ 12 levels "jan","feb","mar",..: 5 5 5 5 5 5 5 5 5 5 ...
##  $ duration : num [1:45211] 261 151 76 92 198 139 217 380 50 55 ...
##  $ campaign : num [1:45211] 1 1 1 1 1 1 1 1 1 1 ...
##  $ pdays    : num [1:45211] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
##  $ previous : num [1:45211] 0 0 0 0 0 0 0 0 0 0 ...
##  $ poutcome : Factor w/ 4 levels "unknown","failure",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ y        : logi [1:45211] FALSE FALSE FALSE FALSE FALSE FALSE ...
```

We have 45211 rows and 16 columns (excluding y).

A look at the description by the researchers tells us that there are no missing values even though some columns have values “unknown”. We have to decide whether to keep them as “unknown” or convert them to NA. Either way, missing values must be inspected.

```
sum(apply(bank_full == "unknown", 1, any))
```

```
## [1] 37369
```

There are a total of 37369 rows with at least one “unknown” value.

How many “unknowns” does each column have?

```
unknown_table <- data.frame(
  unknown_count = sapply(bank_full, function(col) sum(col == "unknown", na.rm = TRUE))) %>%
  arrange(desc(unknown_count)) %>%
  filter(unknown_count != 0)

print(unknown_table)
```

```
##           unknown_count
## poutcome          36959
## contact           13020
## education          1857
## job                 288
```

Almost all of the poutcome values are unknown. Let’s keep this column for now as we will look at outcome distributions with regard to y values later on.

Lastly, since some columns have names that may be difficult to interpret without looking at the metadata first, we should rename them.

```
bank_full <- bank_full %>%
  rename(in_default = "default",
         housing_loan = "housing",
         personal_loan = "loan",
         contact_type = "contact",
         subscribed = "y")
```

```
lapply(bank_full[ , !(names(bank_full) %in% c("age", "balance", "duration", "pdays"))], unique)
```

```
## $job
##  [1] management    technician    entrepreneur  blue-collar   unknown
##  [6] retired       admin         services      self-employed unemployed
## [11] housemaid     student
## 12 Levels: unemployed admin blue-collar entrepreneur housemaid ... unknown
##
## $marital
## [1] married  single   divorced
## Levels: single divorced married
##
## $education
## [1] tertiary  secondary unknown   primary
## Levels: unknown primary secondary tertiary
##
## $in_default
## [1] FALSE  TRUE
##
## $housing_loan
```

```
## [1]  TRUE FALSE
##
## $personal_loan
## [1] FALSE  TRUE
##
## $contact_type
## [1] unknown   cellular  telephone
## Levels: unknown cellular telephone
##
## $day
##  [1] 5  6  7  8  9  12 13 14 15 16 19 20 21 23 26 27 28 29 30 2  3  4  11 17 18
## [26] 24 25 1  10 22 31
## 31 Levels: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 ... 31
##
## $month
##  [1] may jun jul aug oct nov dec jan feb mar apr sep
## Levels: jan feb mar apr may jun jul aug sep oct nov dec
##
## $campaign
##  [1]  1  2  3  5  4  6  7  8  9 10 11 12 13 19 14 24 16 32 18 22 15 17 25 21 43
## [26] 51 63 41 26 28 55 50 38 23 20 29 31 37 30 46 27 58 33 35 34 36 39 44
##
## $previous
##  [1]   0   3   1   4   2  11  16   6   5  10  12   7  18   9  21   8  14  15  26
## [20]  37  13  25  20  27  17  23  38  29  24  51 275  22  19  30  58  28  32  40
## [39]  55  35  41
##
## $poutcome
## [1] unknown failure other   success
## Levels: unknown failure other success
##
## $subscribed
## [1] FALSE  TRUE
```

Looking at the unique columns values we do not see anything out of the ordinary.

Exploratory analysis

Now we can investigate each variable separately.

Subscribed

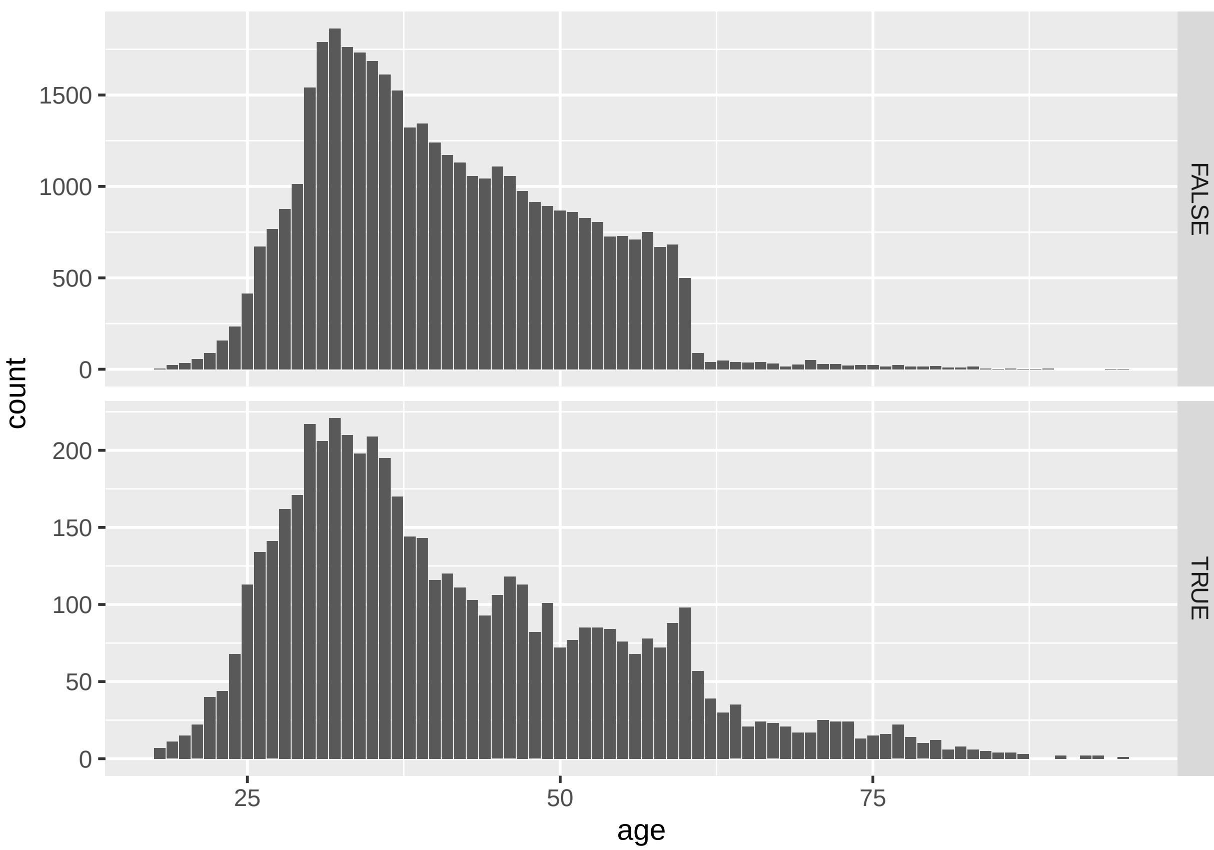
```
table(bank_full$subscribed)
```

```
##
## FALSE  TRUE
## 39922  5289
```

We have a large imbalance in out data. Only 11,6% of contacted clients subscribed. We must also take this into account when removing unknown values.

Age

```
ggplot(bank_full, aes(x = age)) +
  geom_bar() +
  facet_grid(subscribed ~ ., scales = "free_y")
```



The vast majority of clients contacted by the bank were between 25 and 60 years old. Age here is not distributed normally. Using these insights we can create a categorical age variable.

```
bank_full = bank_full %>%
  mutate(age_categ = case_when(
    age > 60 ~ "high",
    age > 25 ~ "mid",
    TRUE ~ "low"
```

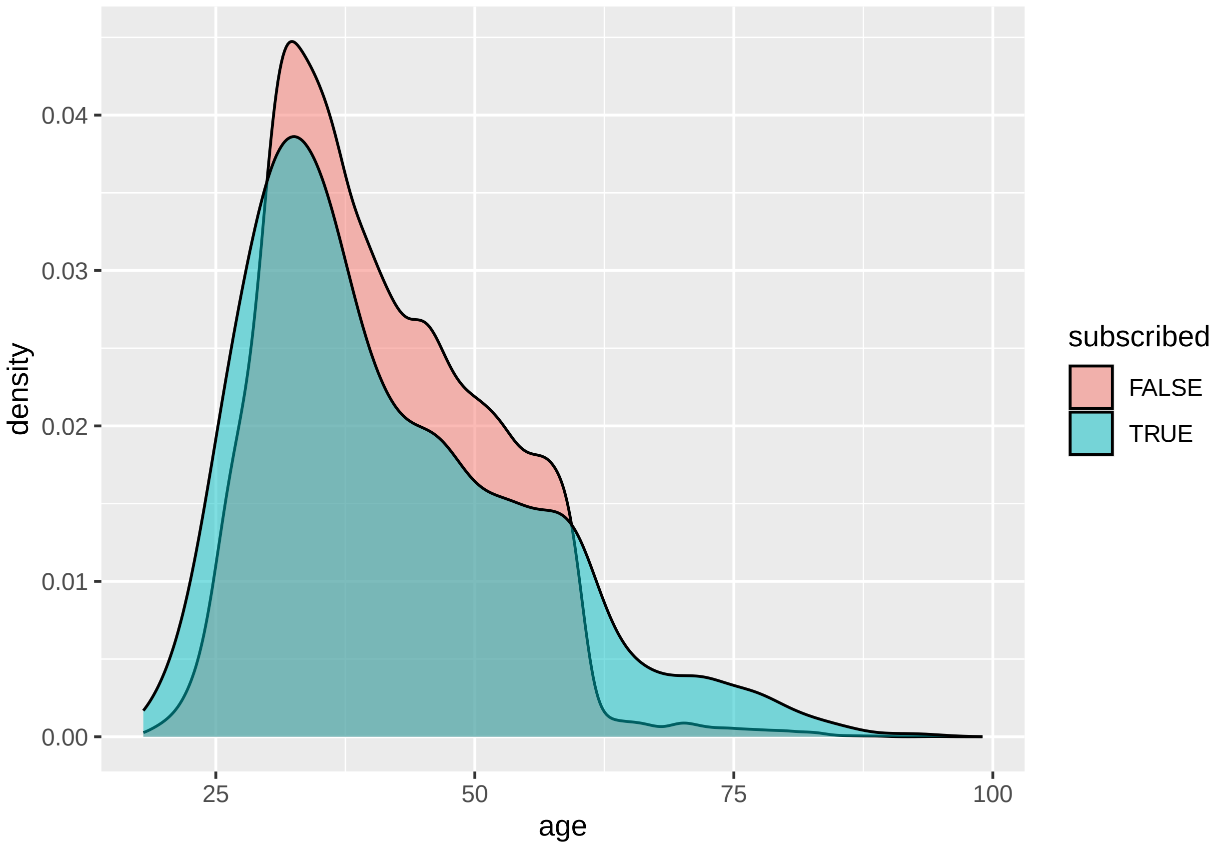
```
  ))

CrossTable(bank_full$subscribed, bank_full$age_categ, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  45211
##
##
##                      | bank_full$age_categ
## bank_full$subscribed |      high |       low |       mid | Row Total |
## ---------------------|-----------|-----------|-----------|-----------|
##                FALSE |       686 |      1016 |     38220 |     39922 |
##                      |     0.017 |     0.025 |     0.957 |     0.883 |
##                      |     0.577 |     0.760 |     0.895 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##                 TRUE |       502 |       320 |      4467 |      5289 |
##                      |     0.095 |     0.061 |     0.845 |     0.117 |
##                      |     0.423 |     0.240 |     0.105 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##         Column Total |      1188 |      1336 |     42687 |     45211 |
##                      |     0.026 |     0.030 |     0.944 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##
##
```

Clients of at least the age of 60 were most likely to subscribe: 42.3% of them chose to do so. That is the highest percentage of all age groups even though older clients make up the smallest part of the total population.

The data (continuous age variable) does not indicate a linear relationship between age and subscription rates. Either way, we will keep a continuous version of the age variable.

```
ggplot(bank_full, aes(x = age, fill = subscribed)) +
  geom_density(alpha = 0.5) +
  xlim(18, 99)
```



The density plots also do not show a large difference in terms of age with the exception being clients over the age of 60.

Job

```
summary(bank_full$job)
```

```
##     unemployed         admin   blue-collar  entrepreneur     housemaid
##           1303          5171          9732          1487          1240
##     management       retired self-employed      services       student
##           9458          2264          1579          4154           938
##     technician       unknown
##           7597           288
```

There are a total of 228 unknown job values. Due to the large number of rows we can afford to drop the "unknowns".

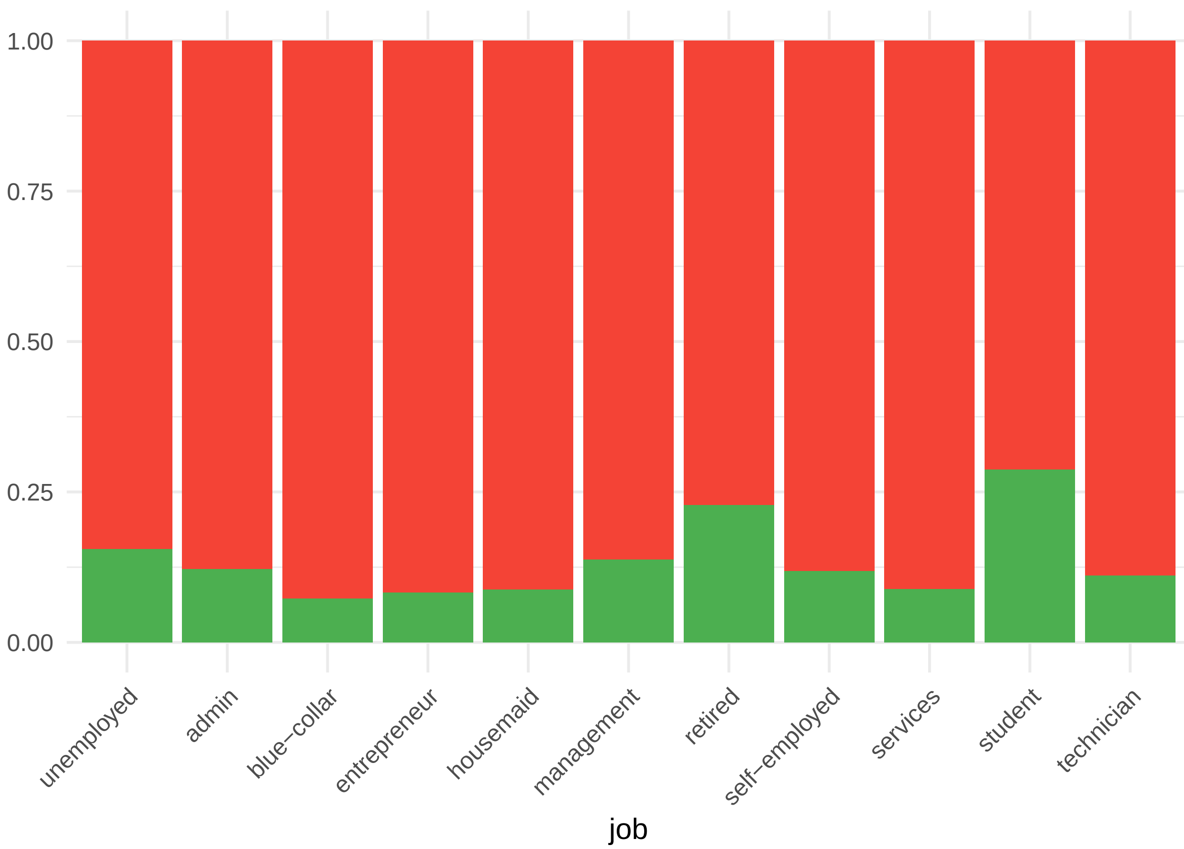
```
bank_full <- bank_full %>% filter(job != "unknown") %>% mutate(job = factor(job))
```

```
nrow(bank_full)
```

```
## [1] 44923
```

Let's look at what percentage of clients subscribed based on their job.

```
create_bar_plot(bank_full, "job")
```



As the chart shows, students, of all jobs, were most likely to subscribe to a deposit (28,7%) with retired workers following second at 22,8%.

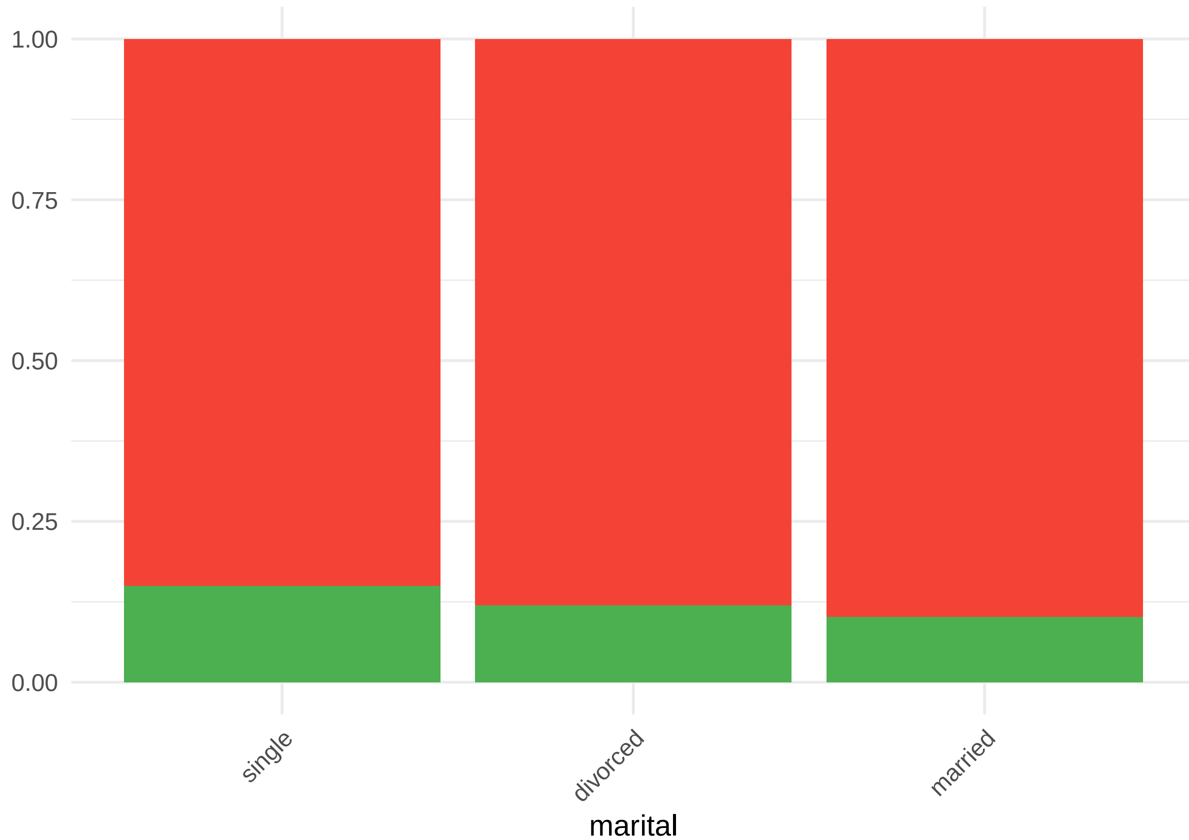
Marital status

```r
CrossTable(bank_full$subscribed, bank_full$marital, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_full$marital
## bank_full$subscribed |    single |  divorced |   married | Row Total |
## ---------------------|-----------|-----------|-----------|-----------|
##                FALSE |     10822 |      4569 |     24277 |     39668 |
##                      |     0.273 |     0.115 |     0.612 |     0.883 |
##                      |     0.851 |     0.880 |     0.899 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##                 TRUE |      1900 |       621 |      2734 |      5255 |
##                      |     0.362 |     0.118 |     0.520 |     0.117 |
##                      |     0.149 |     0.120 |     0.101 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##         Column Total |     12722 |      5190 |     27011 |     44923 |
##                      |     0.283 |     0.116 |     0.601 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##
##
```

Married clients make up 60,1% of out data set. Single clients were slightly more likely to make a subscription (14,9%) than other clients. It is also probable that this tendency is caused by randomness as marital status categories are not divided equally (single - 28,3%, divorced - 11,6% and married - 60,1%).

```r
create_bar_plot(bank_full, "marital")
```


1.00
0.75
0.50
0.25
0.00
single
divorced
married
marital

Education

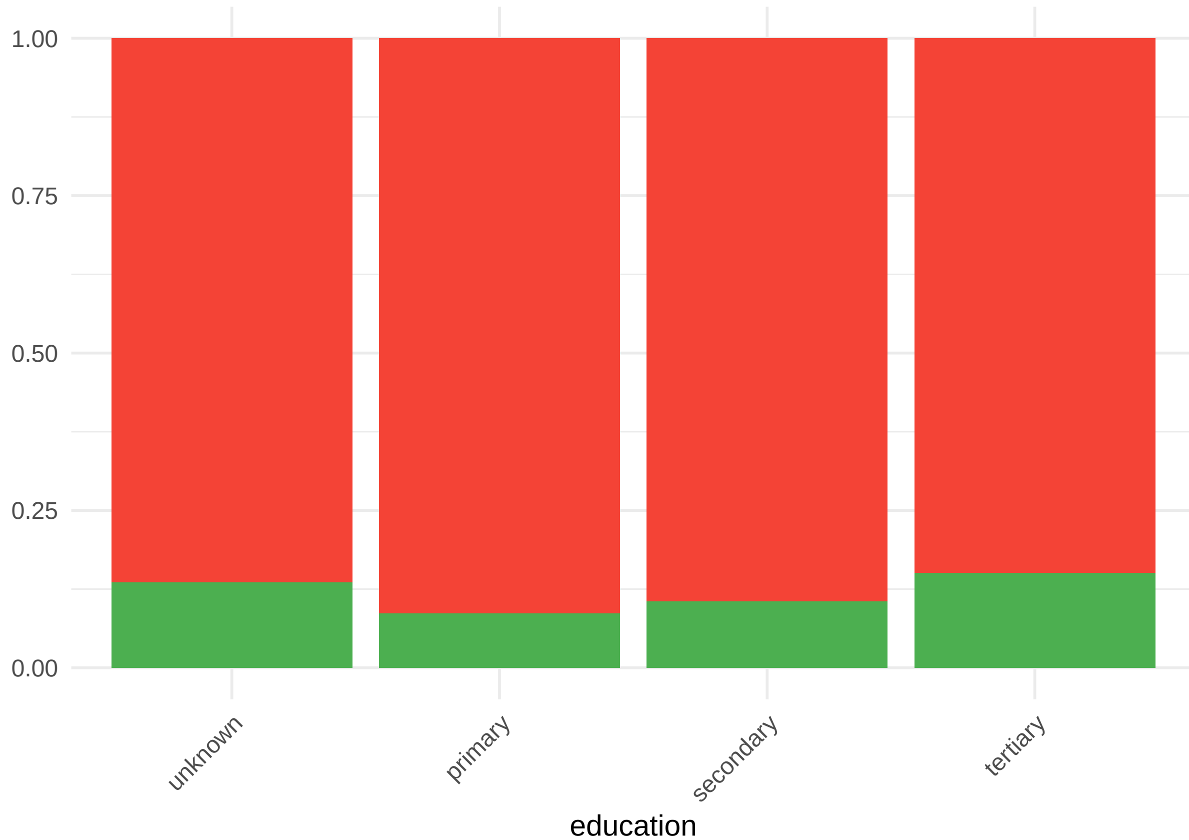
```r
CrossTable(bank_full$subscribed, bank_full$education, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_full$education
## bank_full$subscribed |   unknown |   primary | secondary |  tertiary | Row Total |
## ---------------------|-----------|-----------|-----------|-----------|-----------|
##                FALSE |      1496 |      6212 |     20690 |     11270 |     39668 |
##                      |     0.038 |     0.157 |     0.522 |     0.284 |     0.883 |
##                      |     0.865 |     0.914 |     0.894 |     0.850 |           |
## ---------------------|-----------|-----------|-----------|-----------|-----------|
##                 TRUE |       234 |       588 |      2441 |      1992 |      5255 |
##                      |     0.045 |     0.112 |     0.465 |     0.379 |     0.117 |
##                      |     0.135 |     0.086 |     0.106 |     0.150 |           |
## ---------------------|-----------|-----------|-----------|-----------|-----------|
##         Column Total |      1730 |      6800 |     23131 |     13262 |     44923 |
##                      |     0.039 |     0.151 |     0.515 |     0.295 |           |
## ---------------------|-----------|-----------|-----------|-----------|-----------|
##
##
```

There are 1730 “unknown” values (3,9%) in the education variable. If we removed these “unknowns” we would risk causing further imbalance in the subscribed variable as only 5289 (around 12%) of clients decided to make a deposit subscription in total (234 of them had an “unknown” education).

Clients with a tertiary education (29,5%) are most likely to subscribe out of all groups - 15% of them chose to do so.

```r
create_bar_plot(bank_full, "education")
```


1.00
0.75
0.50
0.25
0.00
unknown
primary
secondary
tertiary
education

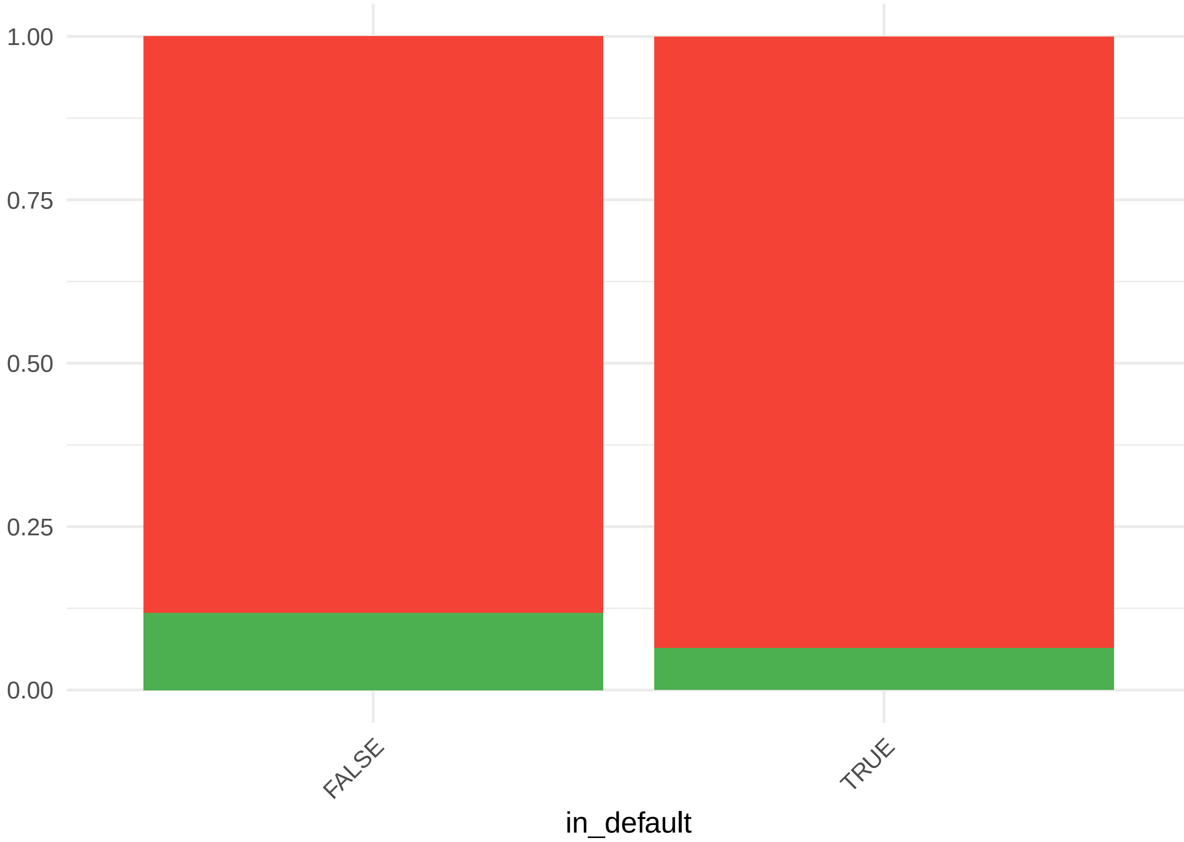
Default status

```r
CrossTable(bank_full$subscribed, bank_full$in_default, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_full$in_default
## bank_full$subscribed |     FALSE |      TRUE | Row Total |
## ---------------------|-----------|-----------|-----------|
##                FALSE |     38907 |       761 |     39668 |
##                      |     0.981 |     0.019 |     0.883 |
##                      |     0.882 |     0.936 |           |
## ---------------------|-----------|-----------|-----------|
##                 TRUE |      5203 |        52 |      5255 |
##                      |     0.990 |     0.010 |     0.117 |
##                      |     0.118 |     0.064 |           |
## ---------------------|-----------|-----------|-----------|
##         Column Total |     44110 |       813 |     44923 |
##                      |     0.982 |     0.018 |           |
## ---------------------|-----------|-----------|-----------|
##
##
```

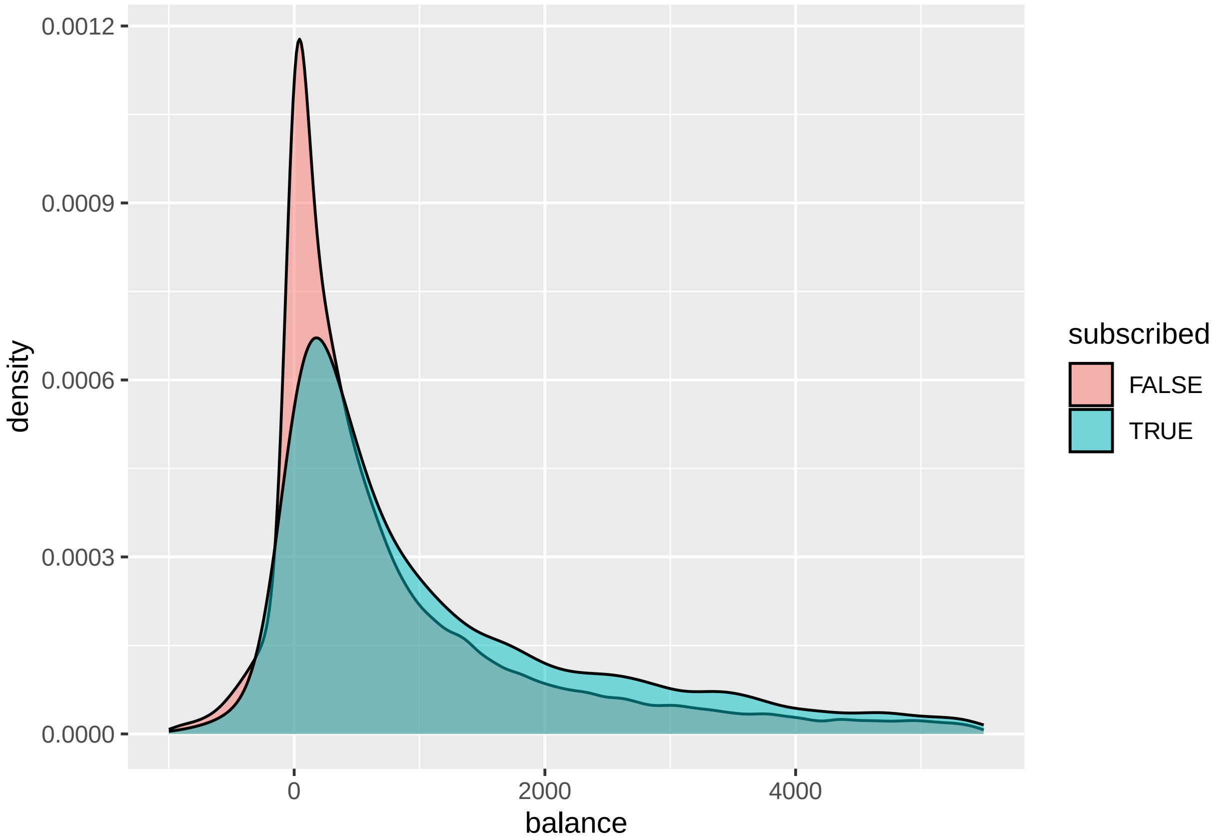
Only 6,4% of clients that were in default chose to make a subscription. Out of the total sample only 1,8% clients were in default. This variable is unlikely to be a good indicator of whether the client makes a subscription.

```r
create_bar_plot(bank_full, "in_default")
```


1.00
0.75
0.50
0.25
0.00
FALSE
TRUE
in_default

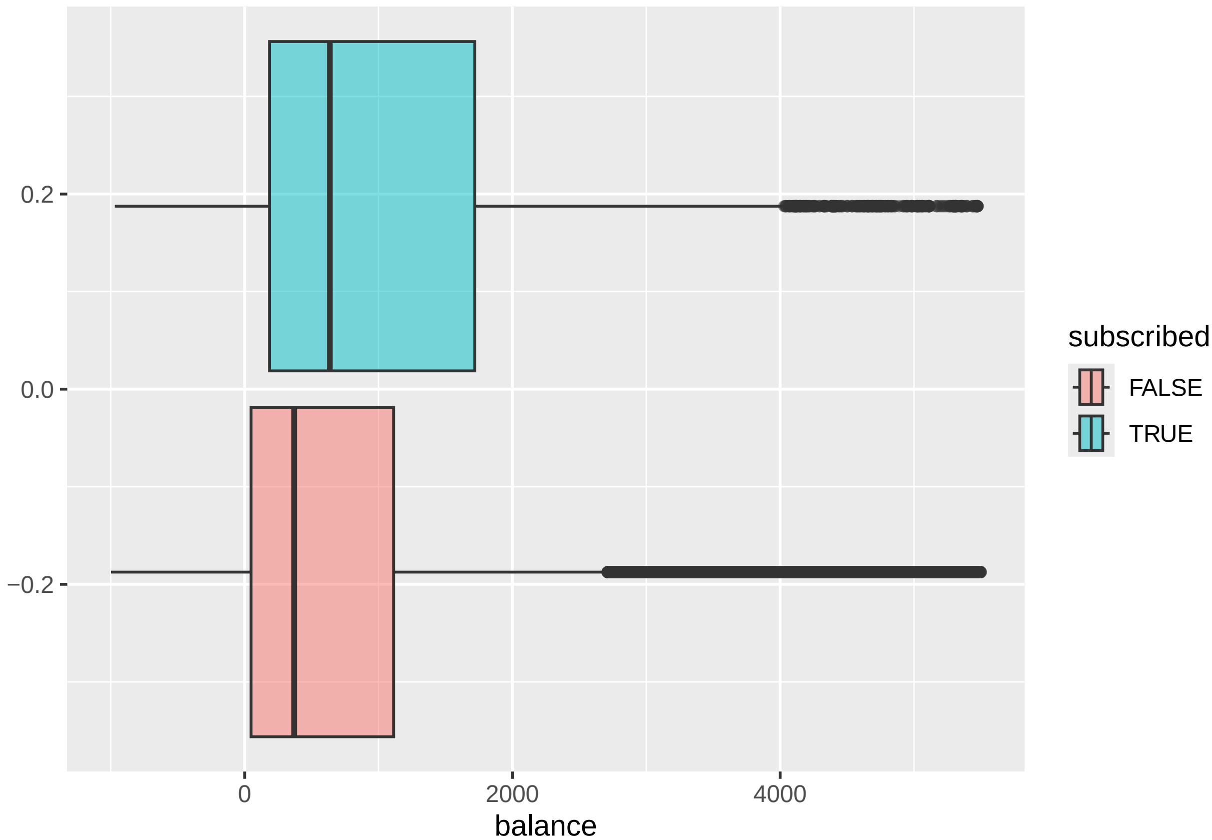
Balance

```
ggplot(bank_full, aes(x = balance, fill = subscribed)) +
  geom_density(alpha = 0.5) +
  xlim(-1000, 5500)
```



The balance density plot does not immediately indicate that wealthier clients are more likely to make a subscription.

```
ggplot(bank_full, aes(x = balance, fill = subscribed)) +
  geom_boxplot(alpha = 0.5) +
  xlim(-1000, 5500)
```



Since we are dealing with financial data, there are many exceptions (outliers) in the distributions of variables. Though the box plots do indicate that the median balance is higher for those who chose to subscribe.

```
paste0("Balance Mean: ", round(mean(bank_full$balance, na.rm = TRUE), 2))
```

```
## [1] "Balance Mean: 1359.64"
```

```
paste0("Balance Standart Deviation: ", round(sd(bank_full$balance), 2))
```

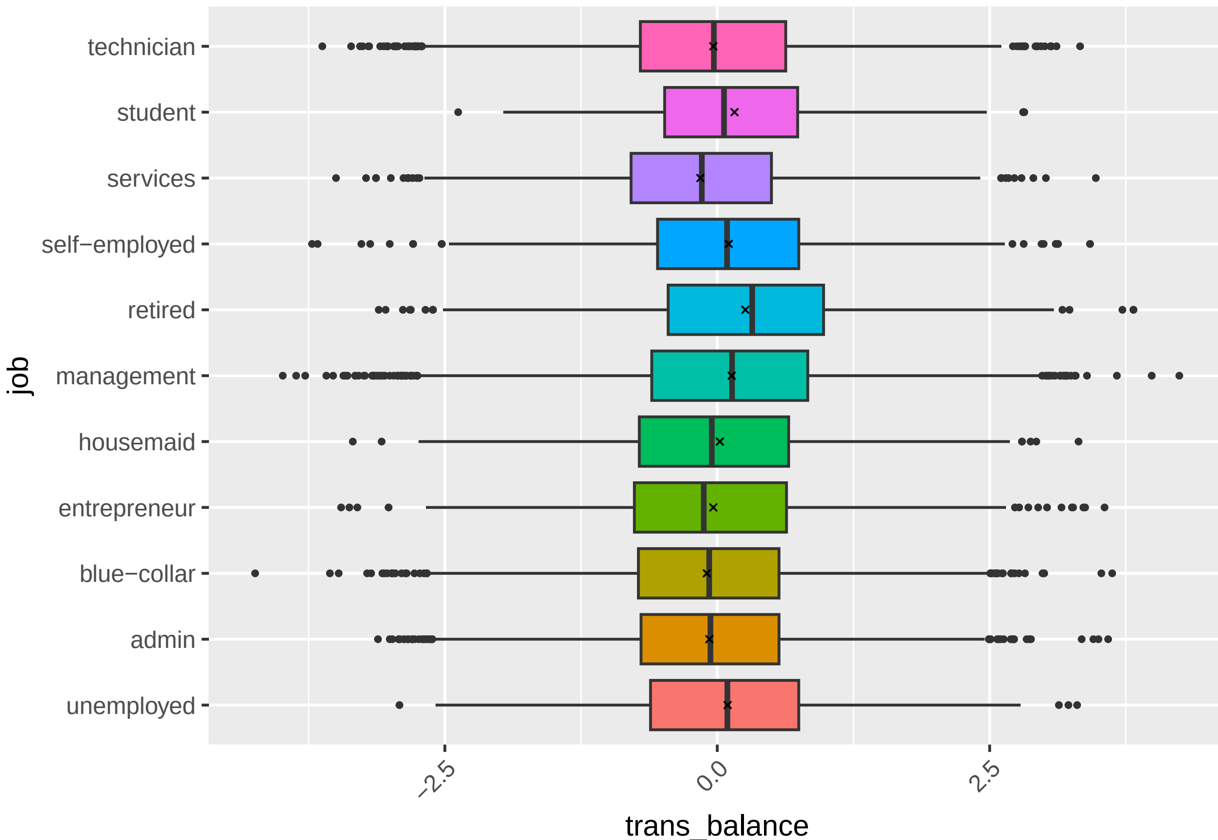
```
## [1] "Balance Standart Deviation: 3045.09"
```

```
outliers <- boxplot.stats(bank_full$balance)$out
outlierNum <- length(outliers)
paste0("Outlier Percentage: ", round(outlierNum/(length(bank_full$balance)) * 100, 2))
```

```
## [1] "Outlier Percentage: 10.49"
```

Since the balance standart deviation is relatively high (3044,77 euros) and 10,49% of the entries can be marked as outliers, we'll normalize the balance variable using the Order-Norm transformation (maps each data point to a percentile in a normal distribution based on the percentile value in the original distribution).

```
on <- orderNorm(bank_full$balance)
bank_full$trans_balance <- predict(on)

ggplot(bank_full, aes(x = job, y = trans_balance, fill = job)) +
  geom_boxplot(outlier.size = 0.7, na.rm = TRUE) +
  coord_flip() +
  stat_summary(fun = mean, geom = "point", shape = 4, size = 0.8, color = "black", na.rm = TRUE) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1), legend.position = "none")
```



The box plots allow us to conclude that the balance of client accounts is likely dependent more factors than simply their job. It also indicates that the clients, grouped by their job type, are not homogeneous (as we had to apply Order-Norm transformation to achieve more normal values). Nevertheless, we can draw certain conclusions. For example, we can see that the median account balance of students is higher than those of service workers. Another trend is clear - retirees have the highest average and median balance.

Housing and Personal loans

```r
CrossTable(bank_full$subscribed, bank_full$housing_loan, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_full$housing_loan
## bank_full$subscribed |     FALSE |      TRUE | Row Total |
## ---------------------|-----------|-----------|-----------|
##                FALSE |     16497 |     23171 |     39668 |
##                      |     0.416 |     0.584 |     0.883 |
##                      |     0.832 |     0.923 |           |
## ---------------------|-----------|-----------|-----------|
##                 TRUE |      3322 |      1933 |      5255 |
##                      |     0.632 |     0.368 |     0.117 |
##                      |     0.168 |     0.077 |           |
## ---------------------|-----------|-----------|-----------|
##         Column Total |     19819 |     25104 |     44923 |
##                      |     0.441 |     0.559 |           |
## ---------------------|-----------|-----------|-----------|
##
##
```

55,9% of the clients in out sample had a housing loan. Clients that did not have a housing loan were more than twice as likely to subscribe than the clients without one. It is clear that this variable will be significant when modelling.

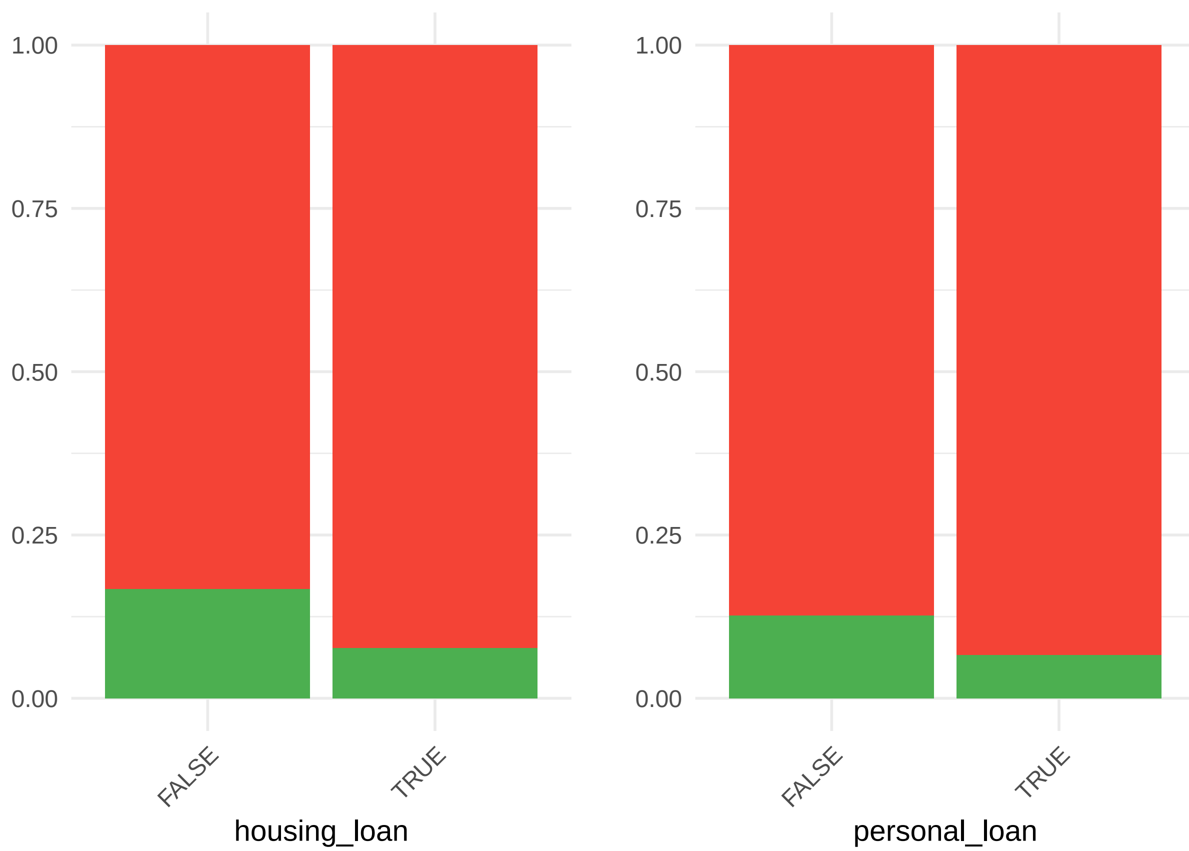
```r
CrossTable(bank_full$subscribed, bank_full$personal_loan, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_full$personal_loan
## bank_full$subscribed |     FALSE |      TRUE | Row Total |
```

```
## ---------------------|-----------|-----------|-----------|
##                FALSE |     32910 |      6758 |     39668 |
##                      |     0.830 |     0.170 |     0.883 |
##                      |     0.873 |     0.933 |           |
## ---------------------|-----------|-----------|-----------|
##                 TRUE |      4773 |       482 |      5255 |
##                      |     0.908 |     0.092 |     0.117 |
##                      |     0.127 |     0.067 |           |
## ---------------------|-----------|-----------|-----------|
##         Column Total |     37683 |      7240 |     44923 |
##                      |     0.839 |     0.161 |           |
## ---------------------|-----------|-----------|-----------|
##
##
```

The situation here is practically the same as with housing loans accept the fact that only 16,1% of the clients had a personal loan. Clients that did not have a personal loan were 1,9 times as likely to subscribe than the clients without one.

```r
plot_list <- lapply(c("housing_loan", "personal_loan"), function(var) create_bar_plot(bank_full, var))

bar_plot_matrix <- grid.arrange(grobs = plot_list, ncol = 2)
```



It is clear that this variable will also be significant when modelling as clients with no financial burdens (defaults and loans) are more likely to subscribe.

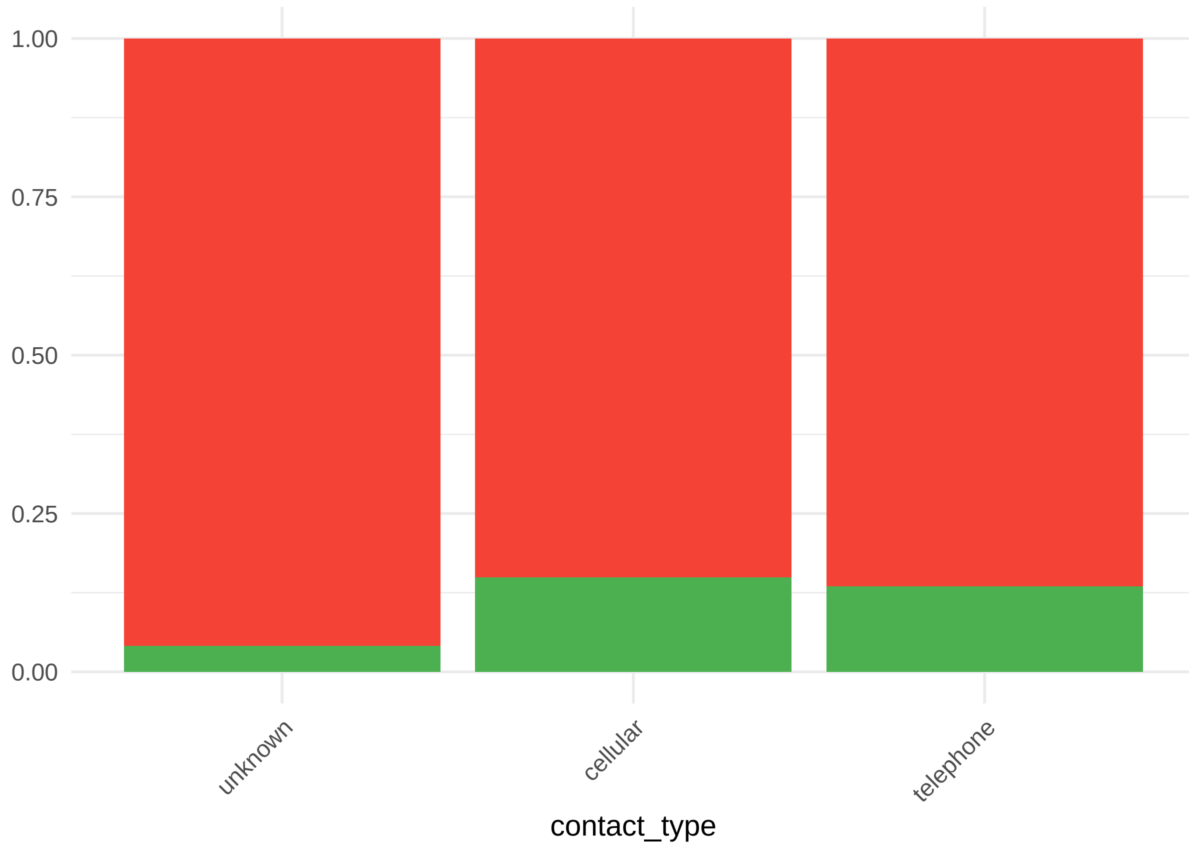
Contact type

```r
CrossTable(bank_full$subscribed, bank_full$contact_type, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_full$contact_type
## bank_full$subscribed |   unknown |  cellular | telephone | Row Total |
## ---------------------|-----------|-----------|-----------|-----------|
##                FALSE |     12381 |     24812 |      2475 |     39668 |
##                      |     0.312 |     0.625 |     0.062 |     0.883 |
##                      |     0.959 |     0.851 |     0.865 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##                 TRUE |       528 |      4342 |       385 |      5255 |
##                      |     0.100 |     0.826 |     0.073 |     0.117 |
##                      |     0.041 |     0.149 |     0.135 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##         Column Total |     12909 |     29154 |      2860 |     44923 |
##                      |     0.287 |     0.649 |     0.064 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##
##
```

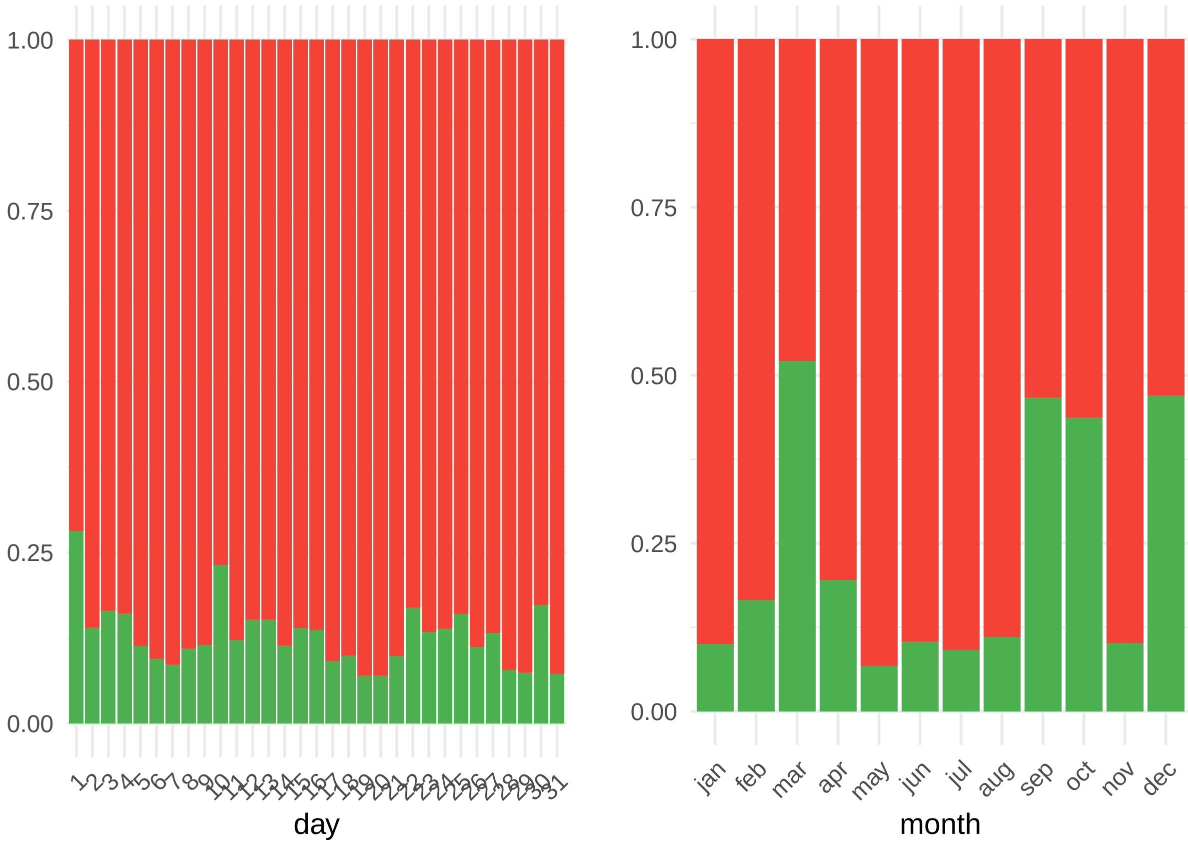
Clients that were contacted through cellular were slightly more likely to make a subscription. The contact type for 28,7% of the clients is unknown.

```r
create_bar_plot(bank_full, "contact_type")
```


1.00
0.75
0.50
0.25
0.00
unknown
cellular
telephone
contact_type

Day and month

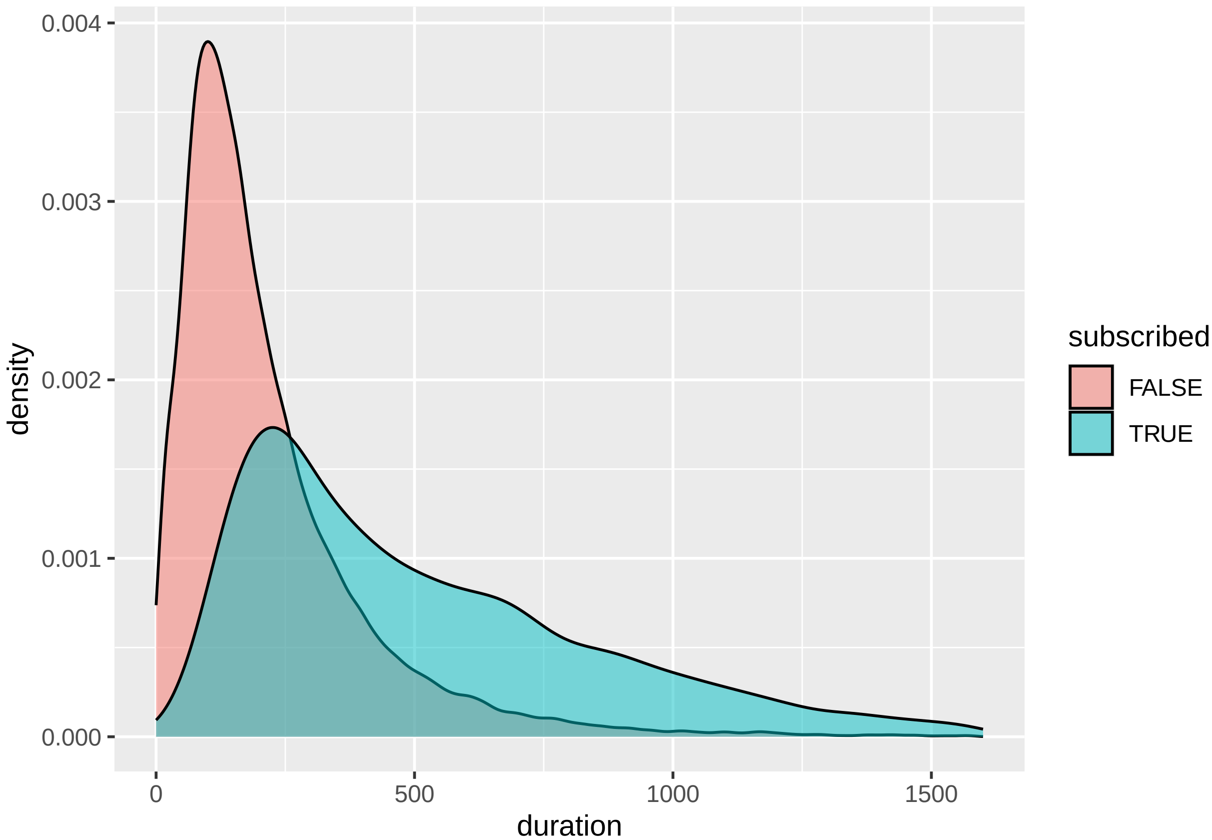
```r
plot_list_2 <- lapply(c("day", "month"), function(var) create_bar_plot(bank_full, var))

bar_plot_matrix_2 <- grid.arrange(grobs = plot_list_2, ncol = 2)
```



March, September, October and December were the best months to contact the clients. Higher success could also be achieved when contacting the clients on the 1st, 10th, 22nd and 30th. These insights should be tested when modelling.

Duration

```r
ggplot(bank_full, aes(x = duration, fill = subscribed)) +
  geom_density(alpha = 0.5) +
  xlim(0, 1600)
```



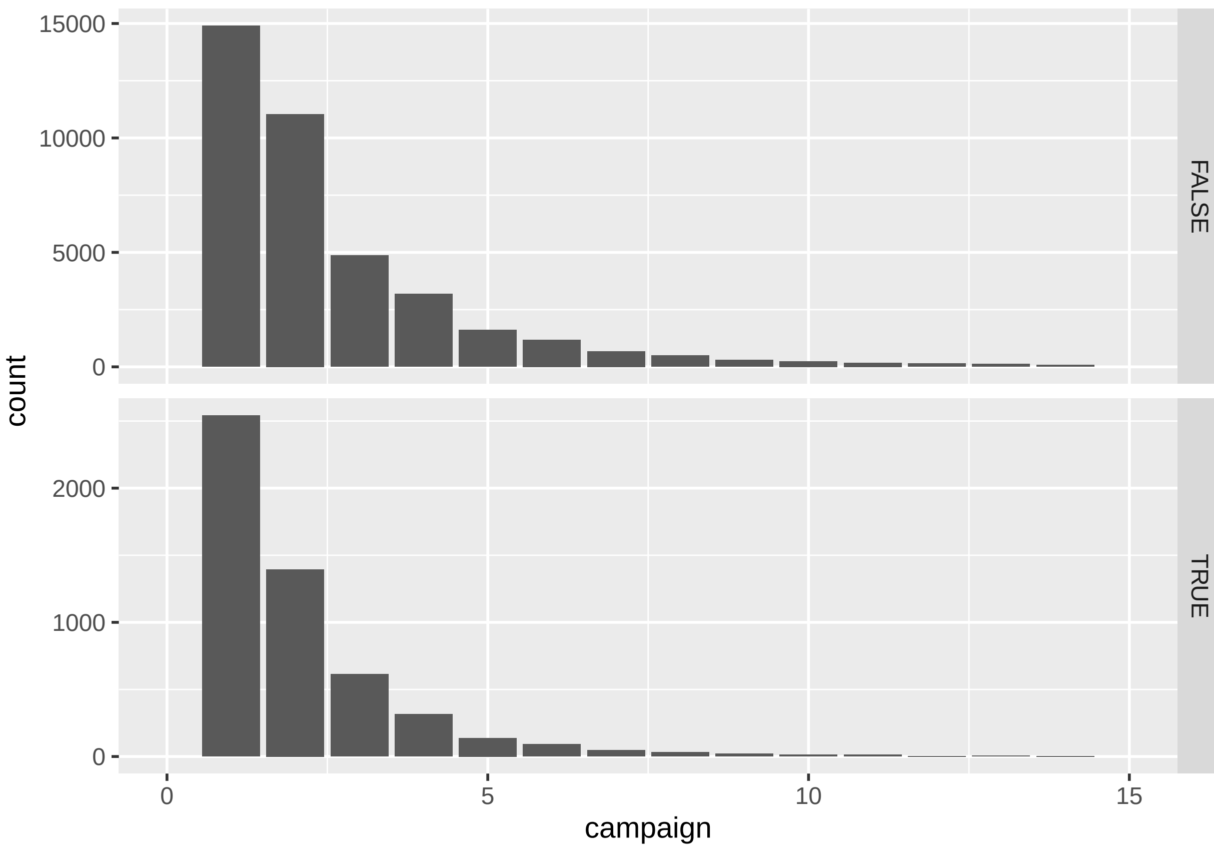
Call duration seems to tell a clearer story than other continuous variables. Clients that, in the end, decided not to subscribe had shorter conversations with the representative of the bank showing their disinterest early on.

Attributes related to previous contact

```r
ggplot(bank_full, aes(x = campaign)) +
  geom_bar() +
  facet_grid(subscribed ~ ., scales = "free_y") +
  xlim(0, 15)
```

Campaign contacts

```
## Warning: Removed 525 rows containing non-finite outside the scale range
## (`stat_count()`).

## Warning: Removed 2 rows containing missing values or values outside the scale range
## (`geom_bar()`).
```




```r
# Warnings were kept on purpose, facet_grid does not knit properly without them
```

Number of contacts performed during this campaign seems to be proportional with the number of contacts performed in total.

Let's look at how the number of total contacts is related to a successful deposit subscription.

```r
subscribed_camp <- bank_full$subscribed[bank_full$campaign < 6]
campaign_camp <- bank_full$campaign[bank_full$campaign < 6]

CrossTable(subscribed_camp, campaign_camp, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
```

```
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  40612
##
##
##                 | campaign_camp
## subscribed_camp |         1 |         2 |         3 |         4 |         5 | Row Total |
## ----------------|-----------|-----------|-----------|-----------|-----------|-----------|
##           FALSE |     14896 |     11043 |      4873 |      3187 |      1610 |     35609 |
##                 |     0.418 |     0.310 |     0.137 |     0.089 |     0.045 |     0.877 |
##                 |     0.854 |     0.888 |     0.888 |     0.910 |     0.921 |           |
## ----------------|-----------|-----------|-----------|-----------|-----------|-----------|
##            TRUE |      2541 |      1395 |       613 |       315 |       139 |      5003 |
##                 |     0.508 |     0.279 |     0.123 |     0.063 |     0.028 |     0.123 |
##                 |     0.146 |     0.112 |     0.112 |     0.090 |     0.079 |           |
## ----------------|-----------|-----------|-----------|-----------|-----------|-----------|
##    Column Total |     17437 |     12438 |      5486 |      3502 |      1749 |     40612 |
##                 |     0.429 |     0.306 |     0.135 |     0.086 |     0.043 |           |
## ----------------|-----------|-----------|-----------|-----------|-----------|-----------|
##
##
```

Number of contacts during the campaign seems to increase the likeliness of subscription but with linearly diminishing returns.

```r
sum(bank_full$pdays != -1)
```

Previous days

```
## [1] 8224
```

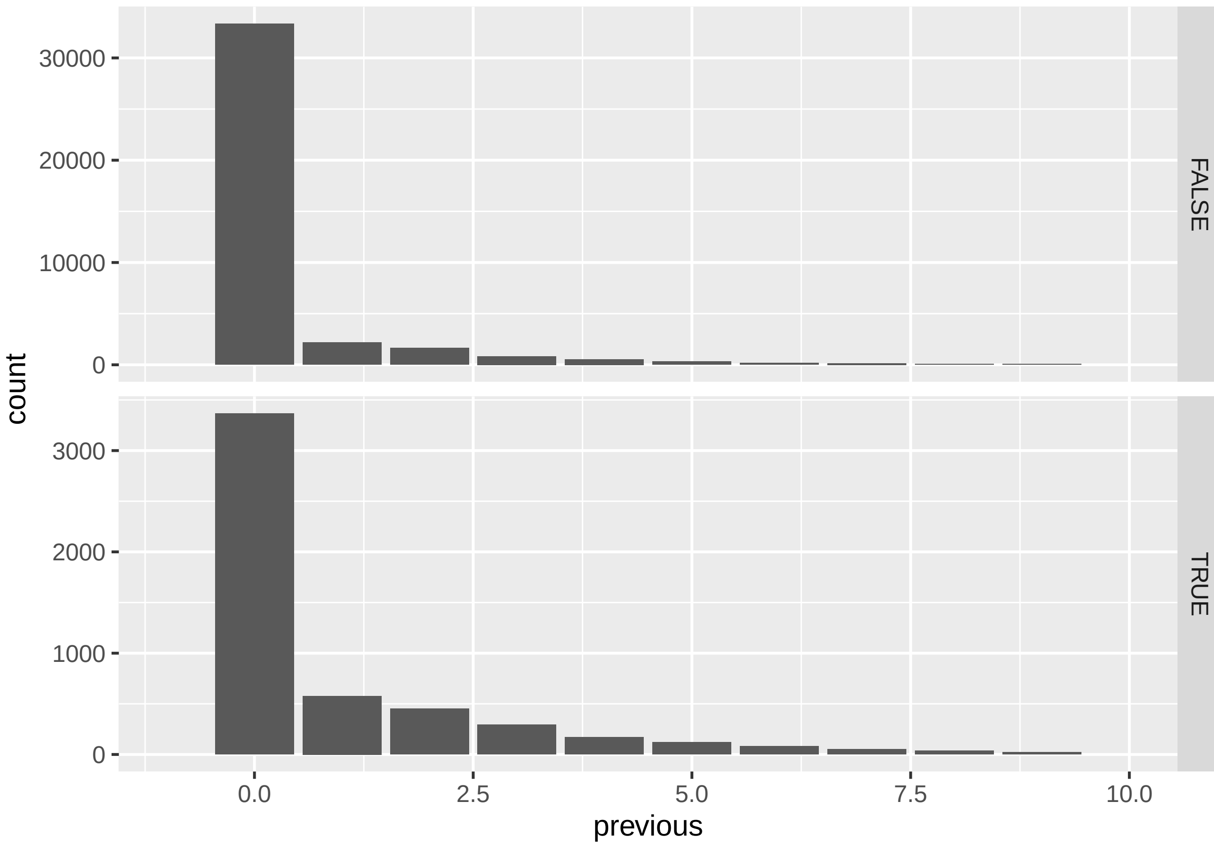
There are 8224 clients which have been contacted in the past. Since there are many different pdays values and because the variable has been encoded as -1 or any other natural number, in order to avoid singularities in our logistic regression model, we can transform this variable in to a binary variable.

```r
bank_full <- bank_full %>%
  mutate(was_contacted = ifelse(pdays == -1, FALSE, TRUE))
```

```r
ggplot(bank_full, aes(x = previous)) +
  geom_bar() +
  facet_grid(subscribed ~ ., scales = "free_y") +
  xlim(-1, 10)
```

Previous contacts

```
## Warning: Removed 294 rows containing non-finite outside the scale range
## (`stat_count()`).

## Warning: Removed 2 rows containing missing values or values outside the scale range
## (`geom_bar()`).
```




```
# Warnings were kept on purpose, facet_grid does not knit properly without them
```

```
subscribed_prev <- bank_full$subscribed[bank_full$previous < 5]
previous_prev <- bank_full$previous[bank_full$previous < 5]

CrossTable(subscribed_prev, previous_prev, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  43407
##
##
##                 | previous_prev
## subscribed_prev |         0 |         1 |         2 |         3 |         4 | Row Total |
## ----------------|-----------|-----------|-----------|-----------|-----------|-----------|
##           FALSE |     33333 |      2184 |      1645 |       847 |       541 |     38550 |
##                 |     0.865 |     0.057 |     0.043 |     0.022 |     0.014 |     0.888 |
##                 |     0.908 |     0.791 |     0.785 |     0.744 |     0.761 |           |
## ----------------|-----------|-----------|-----------|-----------|-----------|-----------|
```

```
##             TRUE |      3366 |       578 |       451 |       292 |       170 |      4857 |
##                  |     0.693 |     0.119 |     0.093 |     0.060 |     0.035 |     0.112 |
##                  |     0.092 |     0.209 |     0.215 |     0.256 |     0.239 |           |
## -----------------|-----------|-----------|-----------|-----------|-----------|-----------|
##     Column Total |     36699 |      2762 |      2096 |      1139 |       711 |     43407 |
##                  |     0.845 |     0.064 |     0.048 |     0.026 |     0.016 |           |
## -----------------|-----------|-----------|-----------|-----------|-----------|-----------|
##
##
```

Number of contacts during the previous campaign seems to linearly increase the likeliness of subscription.

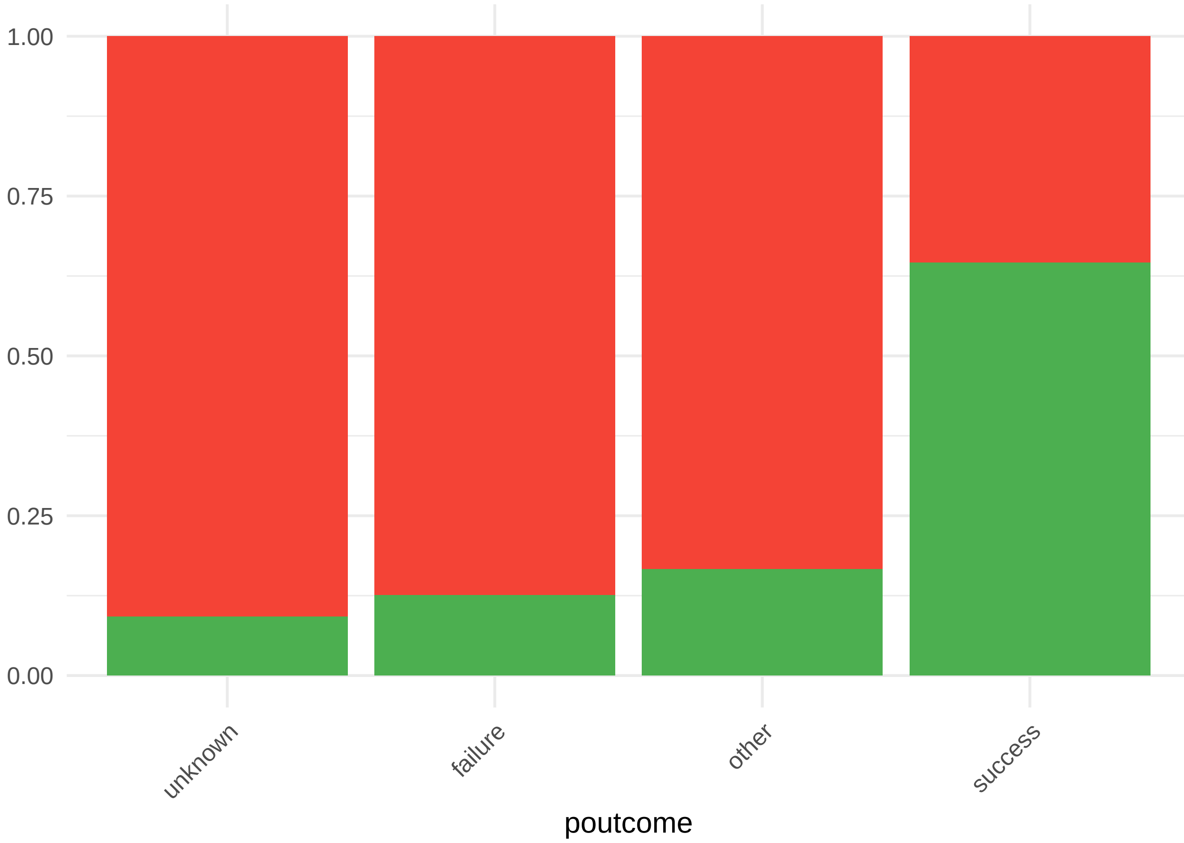
```
CrossTable(bank_full$subscribed, bank_full$poutcome, prop.t = FALSE, prop.chisq = FALSE)
```

Previous outcome

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_full$poutcome
## bank_full$subscribed |   unknown |   failure |     other |   success | Row Total |
## ---------------------|-----------|-----------|-----------|-----------|-----------|
##                FALSE |     33336 |      4269 |      1532 |       531 |     39668 |
##                      |     0.840 |     0.108 |     0.039 |     0.013 |     0.883 |
##                      |     0.908 |     0.875 |     0.834 |     0.354 |           |
## ---------------------|-----------|-----------|-----------|-----------|-----------|
##                 TRUE |      3368 |       612 |       306 |       969 |      5255 |
##                      |     0.641 |     0.116 |     0.058 |     0.184 |     0.117 |
##                      |     0.092 |     0.125 |     0.166 |     0.646 |           |
## ---------------------|-----------|-----------|-----------|-----------|-----------|
##         Column Total |     36704 |      4881 |      1838 |      1500 |     44923 |
##                      |     0.817 |     0.109 |     0.041 |     0.033 |           |
## ---------------------|-----------|-----------|-----------|-----------|-----------|
##
##
```

If the outcome of the previous campaign was successful, the outcome of the current campaign on the same client has a 64,6% likelihood of being successful. Although it must be noted that there are only 1500 clients with the poutcome attribute set as successful.

```r
create_bar_plot(bank_full, "poutcome")
```



Correlation of continuous variables

```
corr_matrix <- cor(bank_full[, c("age", "balance", "duration")], use = "complete.obs")
print(round(corr_matrix, 4))
```

```
##              age balance duration
## age       1.0000  0.0979  -0.0045
## balance   0.0979  1.0000   0.0216
## duration -0.0045  0.0216   1.0000
```

As the continuous variables are not correlated with each other, we can negate multicollinearity concerns for the logistic regression model.

Manipulating data (additional)

We select a small random sample of the provided data with a pre-determined seed for repeatable results.

```
set.seed(167)
smallBank <- sample_n(bank_full, 400, replace = FALSE)
```

Let's choose a data frame with the clients that have a dangerously low balance and have or have had a partner at a point in their life. Due to low numbers in the total population, let's search for them in the full data set.

```
lowBalwPartner <- bank_full %>%
  filter(balance < 100 & marital %in% c("maried", "divorced"))
```

Also, we'll filter another group of clients which have at least one loan with the bank and are at least of the median age for the data set.

```
withLoans <- bank_full %>%
  filter((housing_loan == TRUE | personal_loan == TRUE) & age >= median(age, na.rm = TRUE))
```

We may also calculate the summarizing statistics.

```
job_summary <- bank_full %>%
  group_by(job) %>%
  summarise(
    age_mean = round(mean(age, na.rm = TRUE), 2),
    balance_mean = mean(balance, na.rm = TRUE),
    balance_median = median(balance, na.rm = TRUE),
    balance_sd = sd(balance, na.rm = TRUE),
    duration_median = median(duration, na.rm = TRUE),
    n = n()
  ) %>%
  arrange(desc(n), desc(age_mean))

print(job_summary)
```

```
## # A tibble: 11 x 7
##    job      age_mean balance_mean balance_median balance_sd duration_median     n
##    <fct>       <dbl>        <dbl>          <dbl>      <dbl>           <dbl> <int>
##  1 blue-c~      40.0        1079.            388      2241.             186  9732
##  2 manage~      40.4        1764.            572      3823.             173  9458
##  3 techni~      39.3        1253.            421      2549.             176  7597
##  4 admin        39.3        1136.            396      2642.             174  5171
##  5 servic~      38.7         997.           340.      2164.             186  4154
##  6 retired      61.6        1984.            787      4397.             204  2264
##  7 self-e~      40.5        1648.            526      3684.             179  1579
##  8 entrep~      42.2        1521.            352      4153.             178  1487
##  9 unempl~      41.0        1522.            529      3145.             200  1303
## 10 housem~      46.4        1392.            406      2985.             163  1240
## 11 student      26.5        1388.            502      2442.             180   938
```

The summarized statistics allows us to make a few insights about the clients that were contacted. First, the clients with a job in management had the highest average balance. Second, high standard deviation tells us that client balance varies quite a lot from one client to another. Third, most clients over all had a balance in the mid-500s. Fourth, most of the contacted clients were blue-collar workers. That is quite normal as blue-collar workers usually make up the largest percentage of the population.

We should also inspect the clients that chose to subscribe to a deposit and what characteristics they show.

```r
subscriber_summary <- bank_full %>%
  filter(subscribed == TRUE) %>%
  select(-in_default) %>%
  summarise(across(everything(), ~DescTools::Mode(.x), .names = "mode_{.col}"))

print(subscriber_summary)
```

```
## # A tibble: 1 x 19
##   mode_age mode_job   mode_marital mode_education mode_balance mode_housing_loan
##      <dbl> <fct>      <fct>        <fct>                 <dbl> <lgl>
## 1       32 management married      secondary                 0 FALSE
## # i 13 more variables: mode_personal_loan <lgl>, mode_contact_type <fct>,
## #   mode_day <fct>, mode_month <fct>, mode_duration <dbl>, mode_campaign <dbl>,
## #   mode_pdays <dbl>, mode_previous <dbl>, mode_poutcome <fct>,
## #   mode_subscribed <lgl>, mode_age_categ <chr>, mode_trans_balance <dbl>,
## #   mode_was_contacted <lgl>
```

The data shows us that the "most common" client that chose to subscribe to a deposit is a 32 y.o. married management worker which was contacted via phone in May and the phone call lasted 261 seconds. These could be the key factors which influence the probability of subscription.

Using the previous conclusion, we may create a mock variable that assigns a score of how likely each client is to subscribe to a deposit. In order to give sense to the number representation of the score, we will apply a min-max transformation.

```r
find_engagement <- function(duration, balance, housing_loan, personal_loan, in_default) {
  if(in_default != TRUE){
    score <- duration + 10 * (balance / 1000) - housing_loan * 10 - personal_loan * 20
    if (score < 0){
      return(0)
    } else {
      return(score)
    }
  } else {
    return(0)
  }
}

bank_full <- bank_full %>%
  mutate(engagement_score = mapply(find_engagement, duration, balance, housing_loan, personal_loan, in_
  mutate(engagement_score = round((engagement_score - min(engagement_score, na.rm = TRUE)) /
           (max(engagement_score, na.rm = TRUE) - min(engagement_score, na.rm = TRUE)), 3))
```

In order to detect clients that have no loans and sufficient balance to make a bank term deposit (a. k. a. are "good" potential depositors), but have specifically chosen not to, we will create a new indicator column.

```r
bank_full_potencial <- bank_full %>%
  mutate(potential_client = balance > 1000 & campaign > 0 & previous == 0 &
                            !in_default & !housing_loan & !personal_loan)
summary(bank_full_potencial$potential_client)
```

```
##    Mode   FALSE    TRUE
## logical   39801    5122
```

We can see that to 5227 "potential" clients the marketing campaign hasn't been effective.

Modelling

Next, we have to create dummy variables for categorical columns.

```
dmy <- dummyVars(~ age_categ + was_contacted + job + marital + education + balance + contact_type + day

dummy_data <- data.frame(predict(dmy, newdata = bank_full))

dummy_data <- dummy_data[, setdiff(colnames(dummy_data), c("age_categlow", "was_contactedFALSE", "conta

dummy_full <- cbind(dummy_data, subscribed = bank_full$subscribed, in_default = bank_full$in_default, h
```

We can now separate our original data set into two: training and testing.

```
set.seed(167)
sample_size <- round(0.8 * nrow(dummy_full))

train_indices <- sample(seq_len(nrow(dummy_full)), size = sample_size)

train_dummy <- dummy_full[train_indices, ]
test_dummy  <- dummy_full[-train_indices, ]
```

And finally, we can run the model.

```
model1 <- glm(subscribed ~ ., data = train_dummy, family = binomial)
summary(model1)
```

```
##
## Call:
## glm(formula = subscribed ~ ., family = binomial, data = train_dummy)
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -4.708e+00  3.064e-01 -15.368  < 2e-16 ***
## age_categhigh          2.851e-01  1.562e-01   1.826 0.067865 .
## age_categmid          -6.145e-01  1.045e-01  -5.878 4.15e-09 ***
## was_contactedTRUE      2.055e+00  1.456e+00   1.411 0.158204
## job.admin              1.655e-01  1.267e-01   1.306 0.191427
## job.blue.collar       -3.821e-02  1.249e-01  -0.306 0.759580
## job.entrepreneur      -1.049e-01  1.678e-01  -0.625 0.531946
## job.housemaid         -3.357e-01  1.774e-01  -1.892 0.058461 .
## job.management         2.698e-02  1.229e-01   0.219 0.826274
## job.retired           -5.926e-02  1.504e-01  -0.394 0.693548
## job.self.employed     -1.611e-01  1.560e-01  -1.032 0.301902
## job.services           3.123e-02  1.345e-01   0.232 0.816359
## job.student            3.859e-01  1.580e-01   2.443 0.014570 *
## job.technician         3.612e-02  1.225e-01   0.295 0.768079
## marital.divorced      -6.305e-02  7.245e-02  -0.870 0.384185
## marital.married       -2.212e-01  4.859e-02  -4.553 5.30e-06 ***
## education.primary     -1.707e-01  1.203e-01  -1.419 0.155869
## education.secondary    2.372e-02  1.060e-01   0.224 0.822951
## education.tertiary     2.703e-01  1.108e-01   2.440 0.014707 *
## balance                8.766e-06  5.922e-06   1.480 0.138847
## contact_type.cellular  1.563e+00  8.452e-02  18.493  < 2e-16 ***
## contact_type.telephone 1.327e+00  1.152e-01  11.521  < 2e-16 ***
## day.2                 -8.721e-03  2.095e-01  -0.042 0.966788
## day.3                  1.034e-01  2.113e-01   0.489 0.624588
```

```
## day.4                   1.049e-01  2.048e-01   0.512 0.608383
## day.5                  -1.088e-01  2.053e-01  -0.530 0.596247
## day.6                  -1.358e-01  2.102e-01  -0.646 0.518101
## day.7                  -2.542e-01  2.126e-01  -1.196 0.231689
## day.8                   7.793e-02  2.064e-01   0.378 0.705688
## day.9                   1.296e-01  2.125e-01   0.610 0.541878
## day.10                  7.311e-01  2.330e-01   3.138 0.001699 **
## day.11                 -1.118e-02  2.106e-01  -0.053 0.957661
## day.12                  2.623e-01  2.057e-01   1.275 0.202207
## day.13                  4.665e-01  2.058e-01   2.267 0.023382 *
## day.14                  2.656e-01  2.064e-01   1.287 0.198177
## day.15                  2.633e-01  2.057e-01   1.280 0.200599
## day.16                  1.292e-01  2.090e-01   0.618 0.536399
## day.17                 -5.657e-01  2.103e-01  -2.691 0.007134 **
## day.18                 -5.602e-02  2.050e-01  -0.273 0.784661
## day.19                 -4.229e-01  2.221e-01  -1.904 0.056915 .
## day.20                 -2.905e-01  2.070e-01  -1.403 0.160552
## day.21                  6.921e-02  2.101e-01   0.329 0.741837
## day.22                  2.761e-01  2.187e-01   1.262 0.206788
## day.23                  4.618e-01  2.278e-01   2.027 0.042654 *
## day.24                 -1.539e-01  2.663e-01  -0.578 0.563361
## day.25                  3.608e-01  2.230e-01   1.618 0.105640
## day.26                  9.991e-02  2.324e-01   0.430 0.667312
## day.27                  6.571e-01  2.203e-01   2.982 0.002863 **
## day.28                  1.160e-01  2.199e-01   0.528 0.597775
## day.29                 -2.070e-01  2.254e-01  -0.918 0.358399
## day.30                  5.582e-01  2.065e-01   2.704 0.006856 **
## day.31                  1.416e-01  2.794e-01   0.507 0.612201
## month.feb               9.607e-01  1.590e-01   6.042 1.52e-09 ***
## month.mar               2.704e+00  1.823e-01  14.835  < 2e-16 ***
## month.apr               1.209e+00  1.498e-01   8.071 6.96e-16 ***
## month.may               5.695e-01  1.474e-01   3.863 0.000112 ***
## month.jun               1.627e+00  1.607e-01  10.121  < 2e-16 ***
## month.jul               2.752e-01  1.460e-01   1.885 0.059466 .
## month.aug               4.772e-01  1.473e-01   3.239 0.001201 **
## month.sep               1.898e+00  1.790e-01  10.603  < 2e-16 ***
## month.oct               2.098e+00  1.669e-01  12.572  < 2e-16 ***
## month.nov               5.078e-01  1.592e-01   3.189 0.001427 **
## month.dec               2.016e+00  2.292e-01   8.799  < 2e-16 ***
## campaign               -8.656e-02  1.137e-02  -7.612 2.70e-14 ***
## pdays                  -2.664e-05  3.492e-04  -0.076 0.939194
## previous                6.941e-03  6.291e-03   1.103 0.269865
## poutcome.failure       -1.968e+00  1.454e+00  -1.354 0.175732
## poutcome.other         -1.768e+00  1.455e+00  -1.215 0.224441
## poutcome.success        2.420e-01  1.455e+00   0.166 0.867873
## duration                4.256e-03  7.327e-05  58.082  < 2e-16 ***
## in_defaultTRUE          3.904e-02  1.788e-01   0.218 0.827170
## housing_loanTRUE       -6.243e-01  4.937e-02 -12.646  < 2e-16 ***
## personal_loanTRUE      -4.053e-01  6.758e-02  -5.997 2.01e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 26083  on 35937  degrees of freedom
## Residual deviance: 16838  on 35865  degrees of freedom
## AIC: 16984
##
## Number of Fisher Scoring iterations: 6
```

```r
glm_predict_subs <- predict(model1, test_dummy, type = "response")

roc_curve <- roc(test_dummy$subscribed, glm_predict_subs)

auc(roc_curve)
```

```
## Area under the curve: 0.908
```

The parameters which are Now we remove variables that are not statistically meaningful to the model.

```r
dummy_full_2 <- dummy_full %>%
  select(-c("job.admin", "job.blue.collar", "job.entrepreneur", "job.management",
            "job.retired", "job.self.employed", "job.services", "job.technician",
            "marital.divorced", "education.secondary", "day.2", "day.3", "day.4",
            "day.5", "day.6", "day.8", "day.9", "day.11", "day.12", "day.14",
            "day.15", "day.16", "day.18", "day.20", "day.21", "day.22", "day.24", "day.25",
            "day.26", "day.28", "day.29", "day.31", "campaign", "poutcome.failure",
            "poutcome.other", "poutcome.success", "in_default", "previous", "balance"))

set.seed(167)
sample_size <- round(0.8 * nrow(dummy_full_2))

train_indices_2 <- sample(seq_len(nrow(dummy_full_2)), size = sample_size)

train_dummy_2 <- dummy_full_2[train_indices_2, ]
test_dummy_2  <- dummy_full_2[-train_indices_2, ]
```

```r
model2 <- glm(subscribed ~ ., data = train_dummy_2, family = binomial)
summary(model2)
```

```
##
## Call:
## glm(formula = subscribed ~ ., family = binomial, data = train_dummy_2)
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -4.801e+00  1.815e-01 -26.447  < 2e-16 ***
## age_categhigh             3.422e-01  1.327e-01   2.578 0.009931 **
## age_categmid             -6.426e-01  1.011e-01  -6.353 2.11e-10 ***
## was_contactedTRUE         1.267e+00  8.043e-02  15.747  < 2e-16 ***
## job.housemaid            -3.563e-01  1.371e-01  -2.599 0.009350 **
## job.student               3.801e-01  1.108e-01   3.431 0.000600 ***
## marital.married          -2.282e-01  4.190e-02  -5.446 5.16e-08 ***
## education.primary        -2.803e-01  6.765e-02  -4.144 3.42e-05 ***
## education.tertiary        2.434e-01  4.377e-02   5.562 2.67e-08 ***
## contact_type.cellular     1.600e+00  8.244e-02  19.413  < 2e-16 ***
## contact_type.telephone    1.263e+00  1.122e-01  11.257  < 2e-16 ***
## day.7                    -3.407e-01  1.157e-01  -2.945 0.003234 **
## day.10                    8.568e-01  1.431e-01   5.987 2.14e-09 ***
## day.13                    4.705e-01  9.799e-02   4.801 1.58e-06 ***
```

```
## day.17                    -6.154e-01  1.109e-01  -5.548 2.89e-08 ***
## day.19                    -4.453e-01  1.292e-01  -3.448 0.000566 ***
## day.23                     4.269e-01  1.399e-01   3.052 0.002276 **
## day.27                     5.828e-01  1.260e-01   4.626 3.73e-06 ***
## day.30                     5.092e-01  1.016e-01   5.012 5.38e-07 ***
## month.feb                  9.347e-01  1.354e-01   6.905 5.03e-12 ***
## month.mar                  2.696e+00  1.654e-01  16.297  < 2e-16 ***
## month.apr                  1.197e+00  1.324e-01   9.044  < 2e-16 ***
## month.may                  6.652e-01  1.302e-01   5.110 3.23e-07 ***
## month.jun                  1.681e+00  1.408e-01  11.943  < 2e-16 ***
## month.jul                  3.469e-01  1.311e-01   2.647 0.008120 **
## month.aug                  4.664e-01  1.303e-01   3.579 0.000345 ***
## month.sep                  2.088e+00  1.607e-01  12.999  < 2e-16 ***
## month.oct                  2.206e+00  1.524e-01  14.472  < 2e-16 ***
## month.nov                  4.248e-01  1.381e-01   3.077 0.002089 **
## month.dec                  2.138e+00  2.128e-01  10.048  < 2e-16 ***
## pdays                     -2.036e-03  3.249e-04  -6.268 3.66e-10 ***
## duration                   4.228e-03  7.196e-05  58.757  < 2e-16 ***
## housing_loanTRUE          -7.249e-01  4.730e-02 -15.324  < 2e-16 ***
## personal_loanTRUE         -4.734e-01  6.585e-02  -7.189 6.52e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 26083  on 35937  degrees of freedom
## Residual deviance: 17650  on 35904  degrees of freedom
## AIC: 17718
##
## Number of Fisher Scoring iterations: 6
```

```r
glm_predict_subs2 <- predict(model2, test_dummy_2, type = "response")

roc_curve2 <- roc(test_dummy_2$subscribed, glm_predict_subs2)

auc(roc_curve2)
```

```
## Area under the curve: 0.8958
```

Although, with the statistically insignificant parameters removed, our logistic regression model's AUC is lowered to 0,8958 from 0,908, the model becomes much simpler.

```r
pred_class <- ifelse(glm_predict_subs2 > 0.5, TRUE, FALSE)

confusionMatrix(
  factor(pred_class),
  factor(test_dummy_2$subscribed),
  positive = "TRUE"
)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE TRUE
##      FALSE  7759  719
##      TRUE    211  296
```

```
##
##                Accuracy : 0.8965
##                  95% CI : (0.89, 0.9027)
##     No Information Rate : 0.887
##     P-Value [Acc > NIR] : 0.002206
##
##                   Kappa : 0.3392
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.29163
##             Specificity : 0.97353
##          Pos Pred Value : 0.58383
##          Neg Pred Value : 0.91519
##              Prevalence : 0.11297
##          Detection Rate : 0.03294
##    Detection Prevalence : 0.05643
##       Balanced Accuracy : 0.63258
##
##        'Positive' Class : TRUE
##
```

The sensitivity (true positive) of the model is quite low. Only 29,1% of clients who would subscribe to a deposit are being recognized as "subscribers".

We can try lowering the threshold.

```r
pred_class_2 <- ifelse(glm_predict_subs2 > 0.2, TRUE, FALSE)

confusionMatrix(
  factor(pred_class_2),
  factor(test_dummy_2$subscribed),
  positive = "TRUE"
)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE TRUE
##      FALSE  7218  347
##      TRUE    752  668
##
##                Accuracy : 0.8777
##                  95% CI : (0.8707, 0.8844)
##     No Information Rate : 0.887
##     P-Value [Acc > NIR] : 0.9973
##
##                   Kappa : 0.4802
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.65813
##             Specificity : 0.90565
##          Pos Pred Value : 0.47042
##          Neg Pred Value : 0.95413
##              Prevalence : 0.11297
```

```
##          Detection Rate : 0.07435
##    Detection Prevalence : 0.15804
##       Balanced Accuracy : 0.78189
##
##        'Positive' Class : TRUE
##
```

By lowering the threshold down to 0.2, true positives are being recognized with 65,8% accuracy (up from 29,1%) and the specificity is only lowered to 90,5% (from 97,4%).

Conclusion

1. The logistic regression model accuracy score is 0,8777 (with threshold adjusted). True positive rate is 0,65813.
2. Most important parameters for choosing a potential bank deposit subscriber are call duration, contact type, day and month of contact and whether or not the client has borrowed a loan.