Predictive Bank Client Deposit Rate Analysis

Portfolio Project

Mykolas Motiejūnas

2025-07

Contents

1 Packages

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

2 Custom functions

```r
create_bar_plot <- function(data, var_name) {
  ggplot(data, aes(x = .data[[var_name]], fill = subscribed)) +
    geom_bar(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"
    )
}

create_count_plot <- function(data, var_name) {
  ggplot(data, aes(x = .data[[var_name]], fill = subscribed)) +
    geom_bar() +
    scale_fill_manual(values = c("TRUE" = "#1BC7E4", "FALSE" = "#E4381B")) +
    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"
    )
}

count_unknown <- function(data, value) {
  data %>%
    summarise(across(everything(), ~ sum(.x == value, na.rm = TRUE))) %>%
    pivot_longer(everything(), names_to = "column", values_to = "count") %>%
    filter(count > 0) %>%
    arrange(desc(count))
}

create_model_data <- function(data, exclude_vars = NULL) {
  dummy_vars <- c("age_categ", "was_contacted", "job", "marital", "education",
                  "contact_type", "day", "month", "poutcome")

  dmy <- dummyVars(~ ., data = data[dummy_vars], fullRank = TRUE)
```

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

  model_data <- cbind(
    dummy_data,
    data[c("balance", "campaign", "pdays", "previous",
           "subscribed", "in_default", "housing_loan", "personal_loan")]
  )

  if (!is.null(exclude_vars)) {
    model_data <- model_data %>%
      select(-any_of(exclude_vars))
  }

  return(model_data)
}

train_evaluate_model <- function(data, model_name = "Model", seed = 167) {
  set.seed(seed)

  train_indices <- createDataPartition(data$subscribed, p = 0.8, list = FALSE)
  train_data <- data[train_indices, ]
  test_data <- data[-train_indices, ]

  model <- glm(subscribed ~ ., data = train_data, family = binomial)

  predictions <- predict(model, test_data, type = "response")

  roc_curve <- roc(test_data$subscribed, predictions)
  auc_value <- auc(roc_curve)

  list(
    model = model,
    train_data = train_data,
    test_data = test_data,
    predictions = predictions,
    roc_curve = roc_curve,
    auc = auc_value
  )
}
```

3 Importing data

```r
bank_data <- read.csv("bankData/bank-full.csv",
    sep = ";")
```

4 Data cleaning

```r
head(bank_data)
```

```
##   age          job marital education default balance housing loan contact day
## 1  58   management married  tertiary      no    2143     yes   no unknown   5
## 2  44   technician  single secondary      no      29     yes   no unknown   5
## 3  33 entrepreneur married secondary      no       2     yes  yes unknown   5
## 4  47  blue-collar 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
##   month duration campaign pdays previous poutcome  y
## 1   may      261        1    -1        0  unknown no
## 2   may      151        1    -1        0  unknown no
## 3   may       76        1    -1        0  unknown no
## 4   may       92        1    -1        0  unknown no
## 5   may      198        1    -1        0  unknown no
## 6   may      139        1    -1        0  unknown no
```

```r
tail(bank_data)
```

```
##       age          job  marital education default balance housing loan
## 45206  25   technician   single secondary      no     505      no  yes
## 45207  51   technician  married  tertiary      no     825      no   no
## 45208  71      retired divorced   primary      no    1729      no   no
## 45209  72      retired  married secondary      no    5715      no   no
## 45210  57  blue-collar  married secondary      no     668      no   no
## 45211  37 entrepreneur  married secondary      no    2971      no   no
##         contact day month duration campaign pdays previous poutcome   y
## 45206  cellular  17   nov      386        2    -1        0  unknown yes
## 45207  cellular  17   nov      977        3    -1        0  unknown yes
## 45208  cellular  17   nov      456        2    -1        0  unknown yes
## 45209  cellular  17   nov     1127        5   184        3  success yes
## 45210 telephone  17   nov      508        4    -1        0  unknown  no
## 45211  cellular  17   nov      361        2   188       11    other  no
```

```r
str(bank_data, give.attr = FALSE)
```

```
## 'data.frame':    45211 obs. of  17 variables:
##  $ age      : int  58 44 33 47 33 35 28 42 58 43 ...
##  $ job      : chr  "management" "technician" "entrepreneur" "blue-collar" ...
##  $ marital  : chr  "married" "single" "married" "married" ...
##  $ education: chr  "tertiary" "secondary" "secondary" "unknown" ...
##  $ default  : chr  "no" "no" "no" "no" ...
##  $ balance  : int  2143 29 2 1506 1 231 447 2 121 593 ...
##  $ housing  : chr  "yes" "yes" "yes" "yes" ...
##  $ loan     : chr  "no" "no" "yes" "no" ...
##  $ contact  : chr  "unknown" "unknown" "unknown" "unknown" ...
##  $ day      : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ month    : chr  "may" "may" "may" "may" ...
##  $ duration : int  261 151 76 92 198 139 217 380 50 55 ...
##  $ campaign : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ pdays    : int  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
##  $ previous : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ poutcome : chr  "unknown" "unknown" "unknown" "unknown" ...
```

```
##  $ y        : chr  "no" "no" "no" "no" ...
```

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.

```r
factor_cols <- c("marital", "job", "education", "contact", "poutcome", "day")
logical_cols <- c("default", "housing", "loan", "y")
months <- c("jan", "feb", "mar", "apr", "may", "jun",
            "jul", "aug", "sep", "oct", "nov", "dec")

bank_data <- bank_data %>%
  mutate(
    job = if_else(job == "admin.", "admin", job),
    across(all_of(factor_cols), as.factor),
    across(all_of(logical_cols), ~ .x == "yes"),
    month = factor(month, levels = months),
    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_data, give.attr = FALSE)
```

```
## 'data.frame':    45211 obs. of  17 variables:
##  $ age      : int  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  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ balance  : int  2143 29 2 1506 1 231 447 2 121 593 ...
##  $ housing  : logi  TRUE TRUE TRUE TRUE FALSE TRUE ...
##  $ loan     : logi  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 : int  261 151 76 92 198 139 217 380 50 55 ...
##  $ campaign : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ pdays    : int  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
##  $ previous : int  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  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.

```r
sum(rowSums(bank_data == "unknown") > 0)
```

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

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

How many “unknowns” does each column have?

```r
count_unknown(bank_data, "unknown")
```

```
## # A tibble: 4 x 2
##   column    count
##   <chr>     <int>
## 1 poutcome  36959
## 2 contact   13020
## 3 education  1857
## 4 job         288
```

Almost all of the poutcome column values are unknown. Let's keep this column for now as we will look at outcome distributions with regard to y (subscription outcome) 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.

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

```r
lapply(bank_data[ , !(names(bank_data) %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.

5 Exploratory analysis

Now we can investigate each variable separately.

5.1 Subscribed

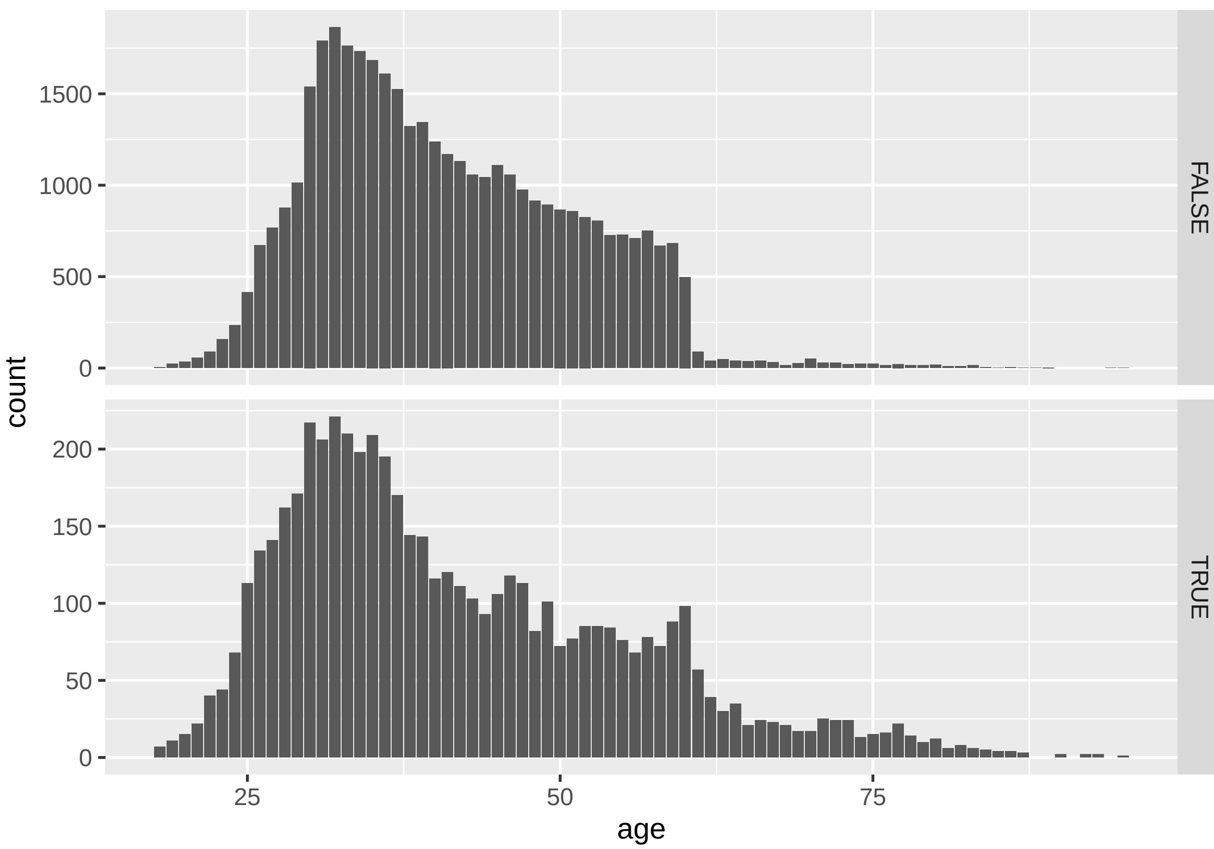
```
table(bank_data$subscribed)
```

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

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

5.2 Age

```
ggplot(bank_data, 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_data <- bank_data %>%
  mutate(
    age_categ = case_when(age > 60 ~ "high", age > 25 ~ "mid", TRUE ~ "low"),
    age_categ = factor(age_categ),
```

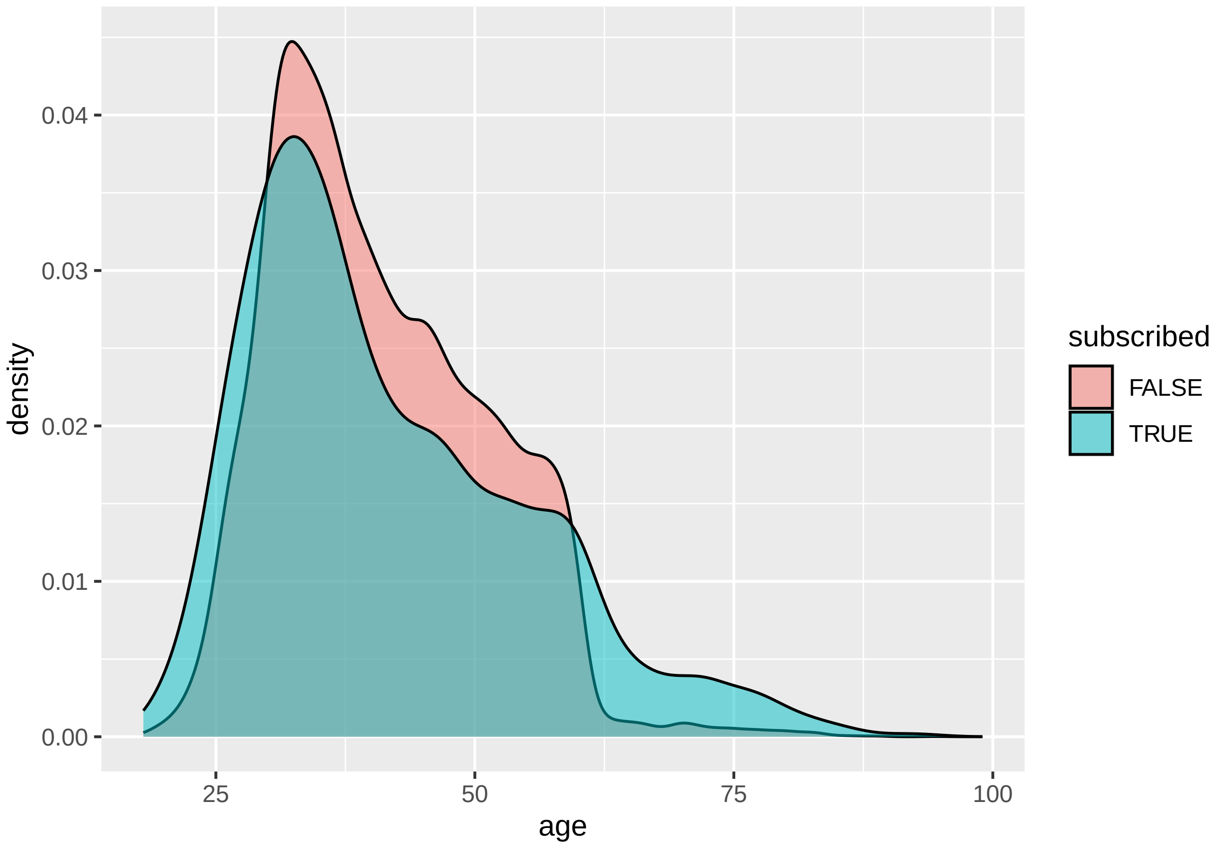
```r
    age_categ = relevel(age_categ, ref = "low")
  )

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

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  45211
##
##
##                      | bank_data$age_categ
## bank_data$subscribed |       low |      high |       mid | Row Total |
## ---------------------|-----------|-----------|-----------|-----------|
##                FALSE |      1016 |       686 |     38220 |     39922 |
##                      |     0.025 |     0.017 |     0.957 |     0.883 |
##                      |     0.760 |     0.577 |     0.895 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##                 TRUE |       320 |       502 |      4467 |      5289 |
##                      |     0.061 |     0.095 |     0.845 |     0.117 |
##                      |     0.240 |     0.423 |     0.105 |           |
## ---------------------|-----------|-----------|-----------|-----------|
##         Column Total |      1336 |      1188 |     42687 |     45211 |
##                      |     0.030 |     0.026 |     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 continuous age variable does not indicate a linear relationship between age and subscription rates.

```r
ggplot(bank_data, 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.

5.3 Job

```
summary(bank_data$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 total rows we can afford to drop the "unknowns".

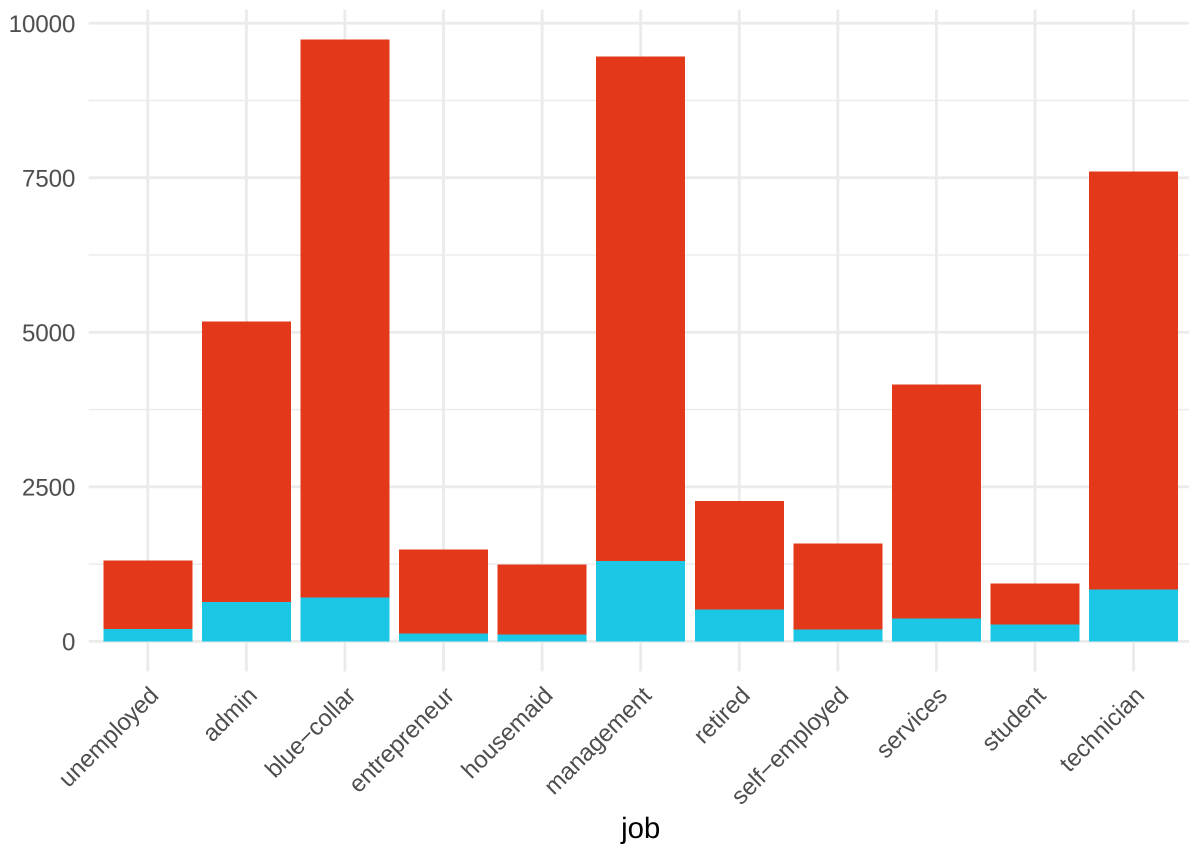
```
bank_data <- bank_data %>%
  filter(job != "unknown") %>%
  mutate(job = factor(job))
```

```
nrow(bank_data)
```

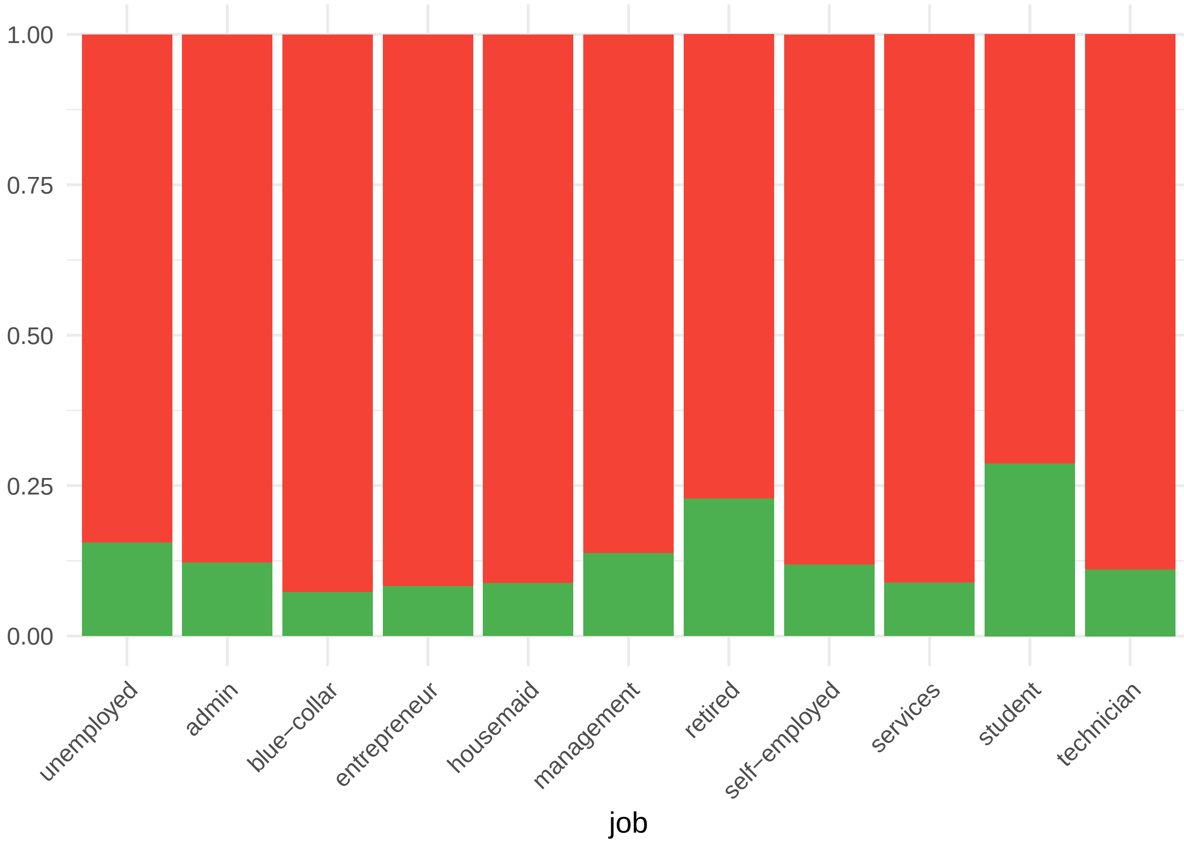
```
## [1] 44923
```

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

```
create_count_plot(bank_data, "job")
```



```r
create_bar_plot(bank_data, "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%. This could be to students having fewer major expenses, such as mortgages or car loans, and being heavily dependent on their parents. Retirees also often seek low-risk investment options, making bank deposits attractive.

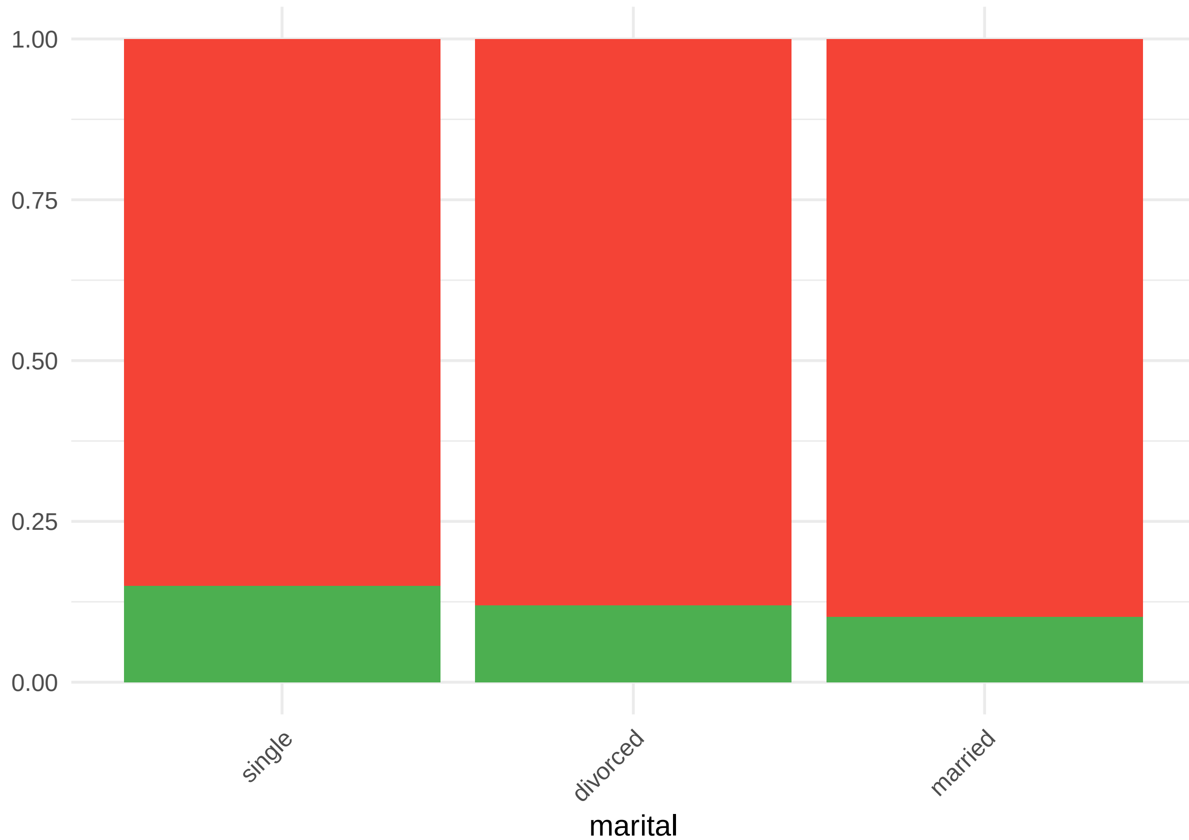
5.4 Marital status

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

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_data$marital
## bank_data$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 our data set. Single clients were slightly more likely to subscribe to a deposit (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%).

```
create_bar_plot(bank_data, "marital")
```


1.00
0.75
0.50
0.25
0.00
single
divorced
married
marital

5.5 Education

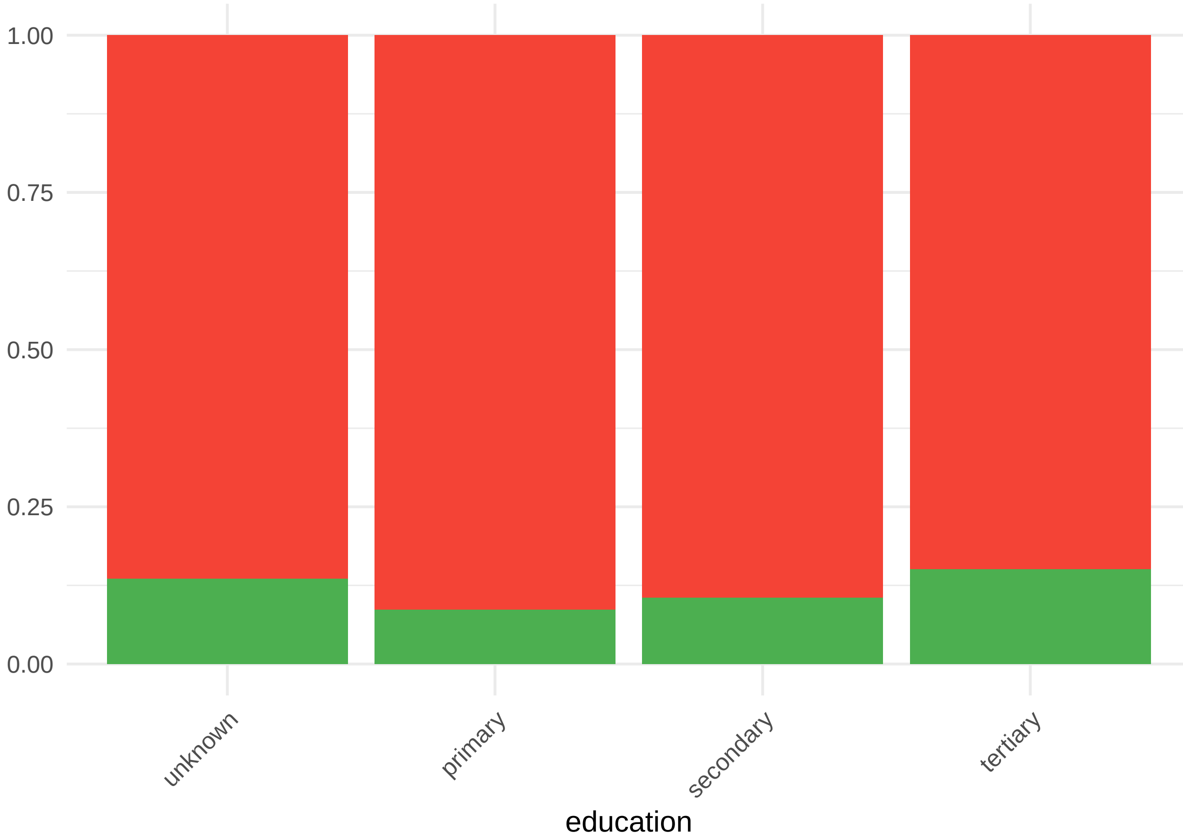
```
CrossTable(bank_data$subscribed, bank_data$education, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_data$education
## bank_data$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 5255 (11,7%) of clients decided to make a deposit subscription in total (234 of them had an "unknown" education).

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

```
create_bar_plot(bank_data, "education")
```


1.00
0.75
0.50
0.25
0.00
unknown
primary
secondary
tertiary
education

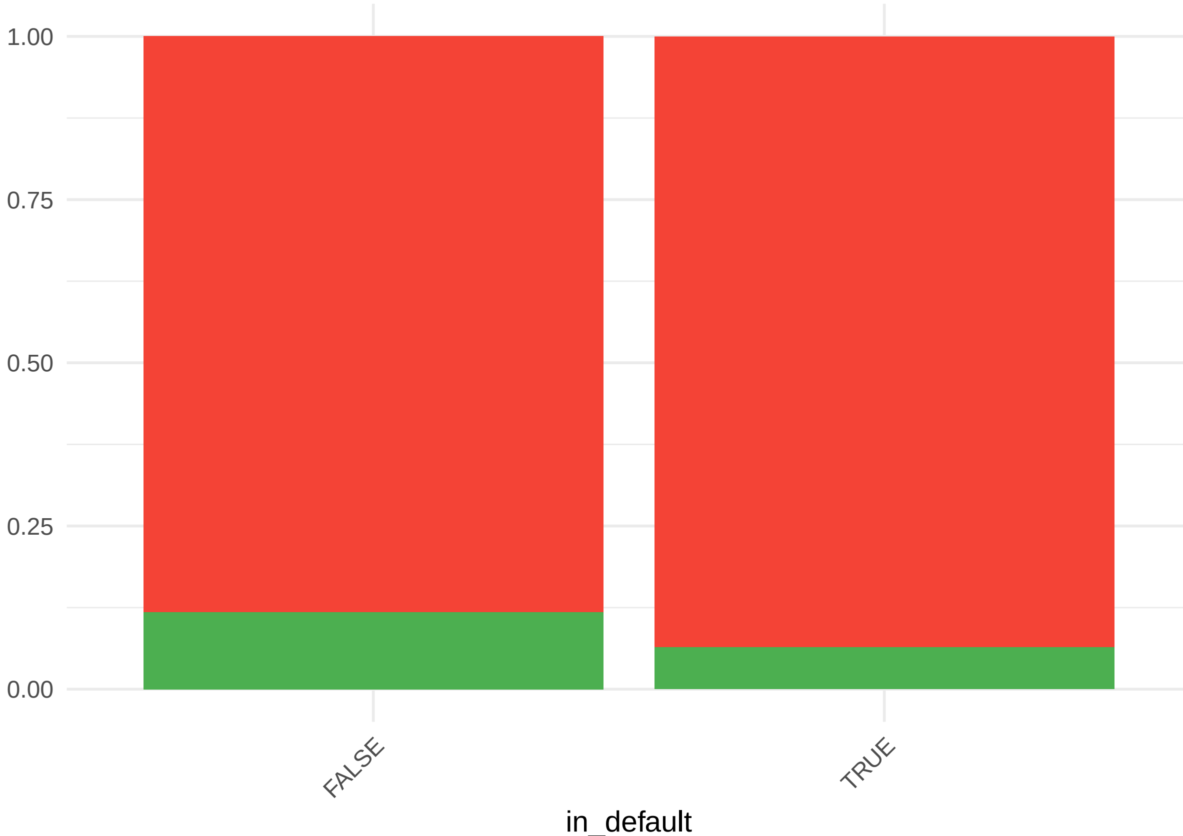
5.6 Default status

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

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_data$in_default
## bank_data$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 |           |
## ---------------------|-----------|-----------|-----------|
##
##
```

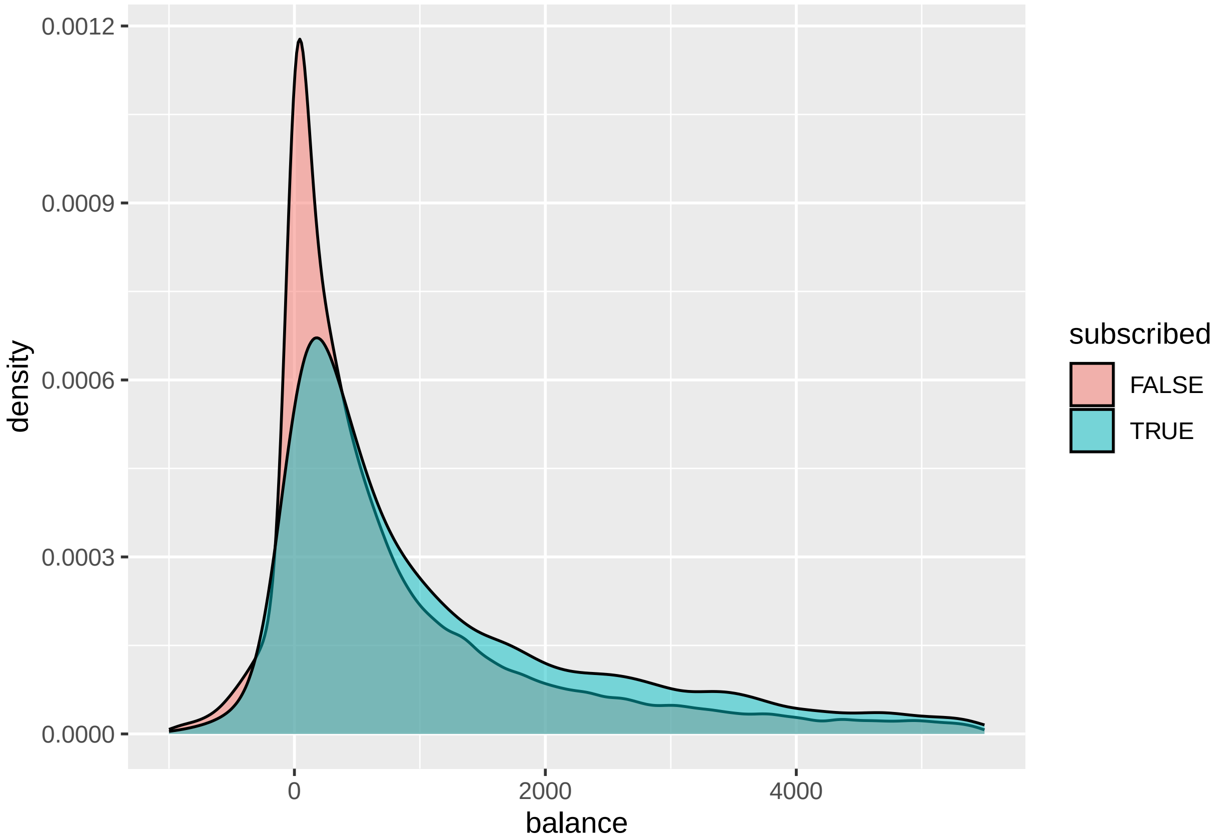
Only 6,4% of clients that were in default chose to sign up for a deposit. Out of the total sample only 1,8% of clients were in default. This variable is unlikely to be a good indicator of whether the client subscribes to a deposit.

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


1.00
0.75
0.50
0.25
0.00
FALSE
TRUE
in_default

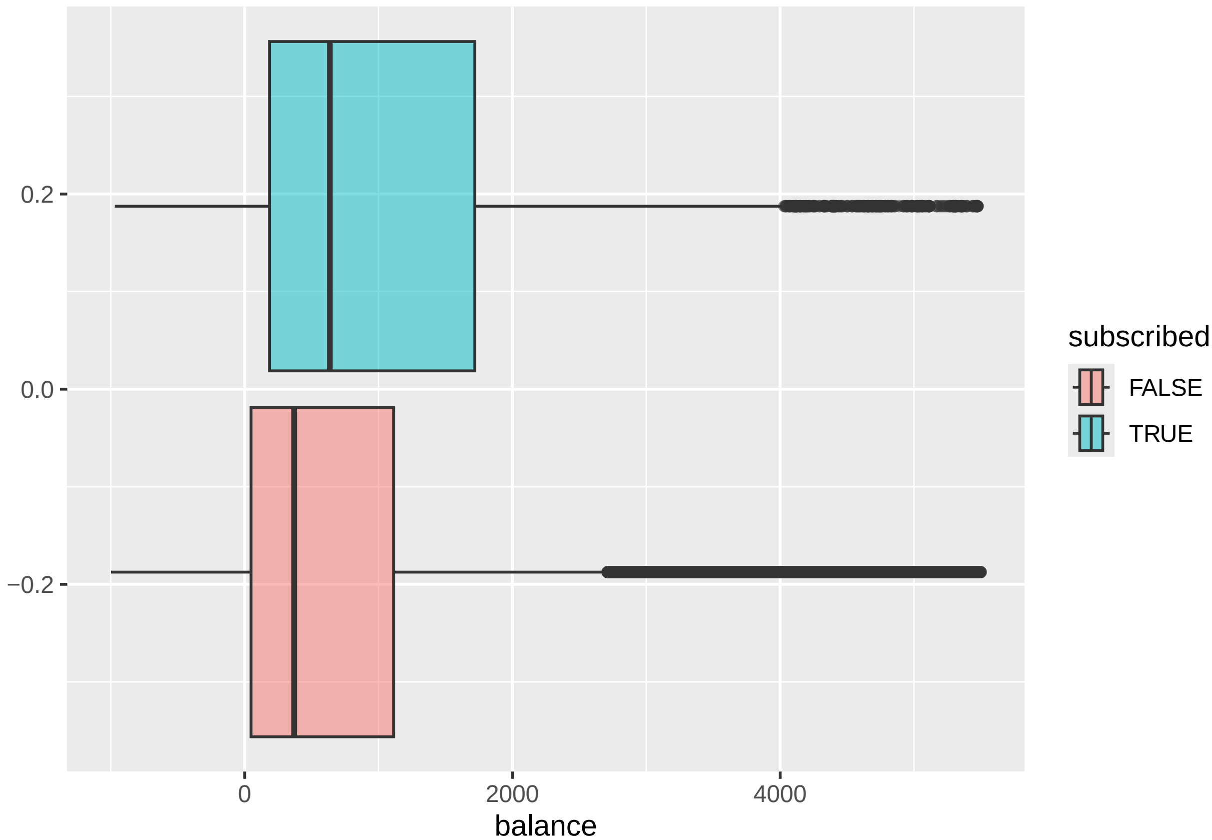
5.7 Balance

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



Since we are dealing with financial data, there are many extreme 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_data$balance, na.rm = TRUE), 2))
```

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

```
paste0("Balance Standard Deviation: ", round(sd(bank_data$balance), 2))
```

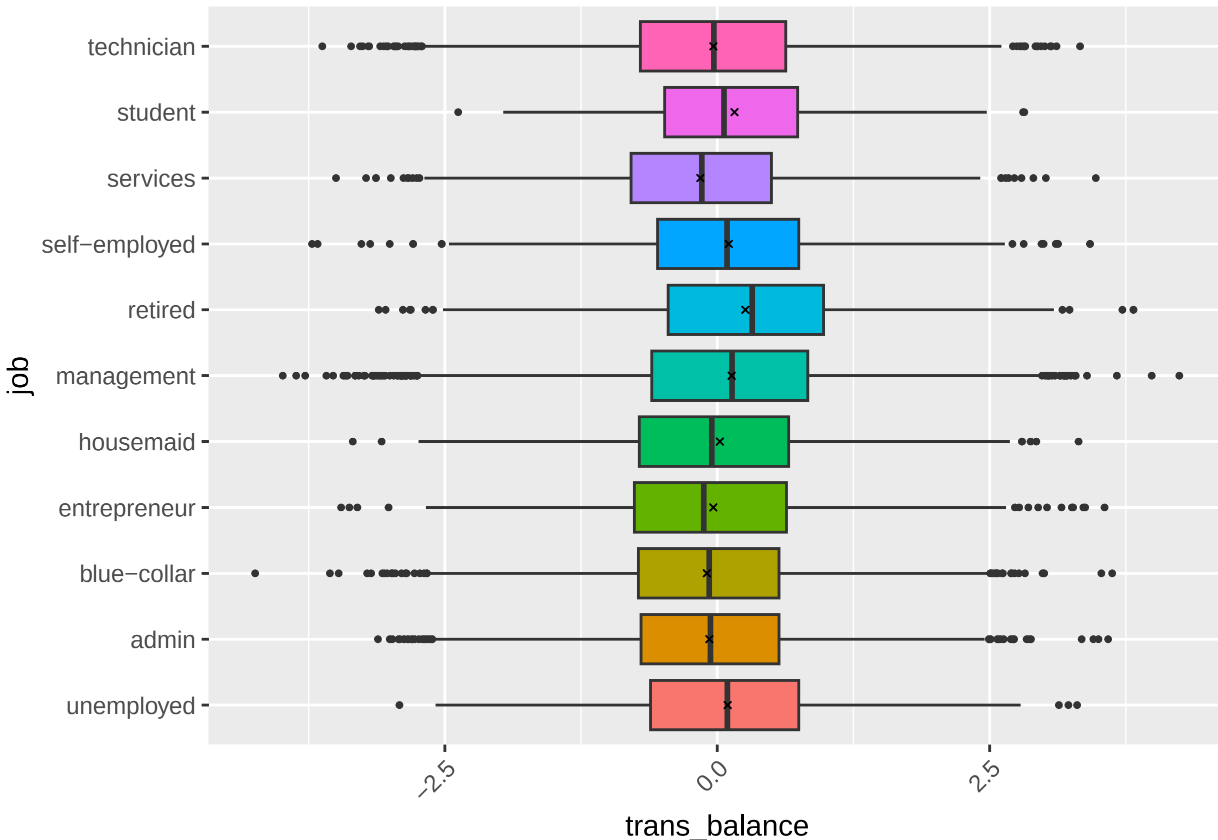
```
## [1] "Balance Standard Deviation: 3045.09"
```

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

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

Since the balance standard deviation is relatively high (3045,09 euros) and 10,49% of the entries can be marked as outliers, we'll normalize the balance variable using the Order-Norm transformation (converts each value to its percentile rank in the original distribution, then maps that percentile to the corresponding value in a standard normal distribution).

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

ggplot(bank_data, 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.

5.8 Housing and Personal loans

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

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_data$housing_loan
## bank_data$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 our 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.

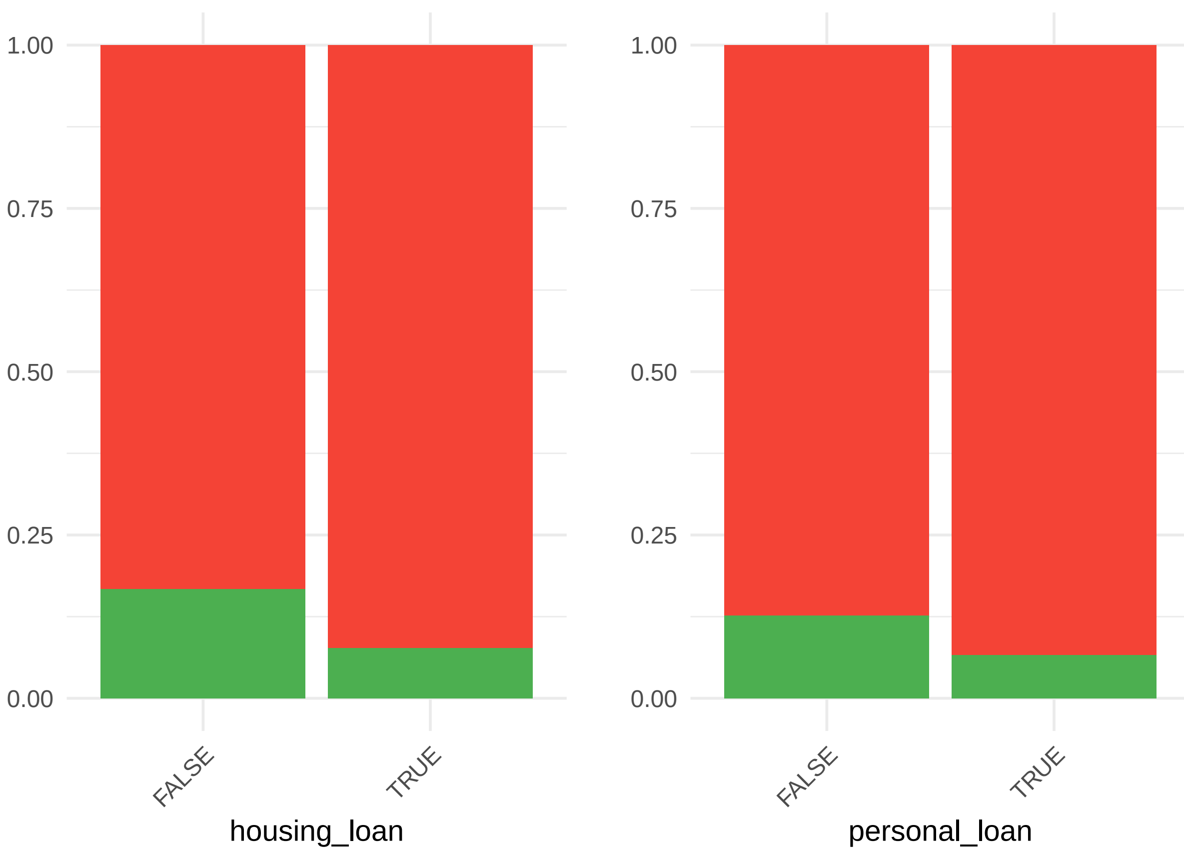
```
CrossTable(bank_data$subscribed, bank_data$personal_loan, prop.t = FALSE, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_data$personal_loan
## bank_data$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 with personal loans is practically the same as with housing loans accept the fact that only 16,1% of the clients had a personal loan (55,9% had a housing loan). Clients that did not have a personal loan were 1,9 times as likely to subscribe than the clients without one.

```
plot_list <- lapply(c("housing_loan", "personal_loan"), function(var) create_bar_plot(bank_data, 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.

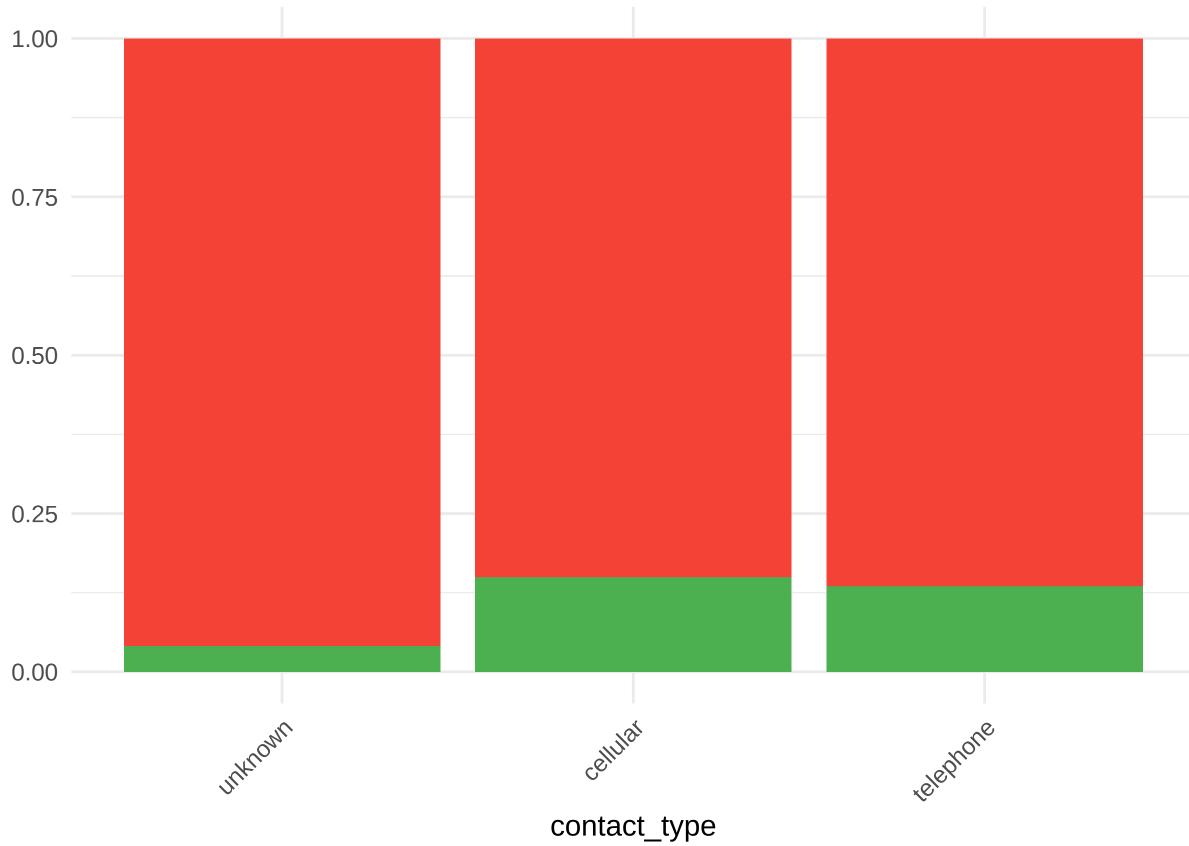
5.9 Contact type

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

```
## 
## 
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
## 
## 
## Total Observations in Table:  44923 
## 
## 
##                      | bank_data$contact_type 
## bank_data$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.

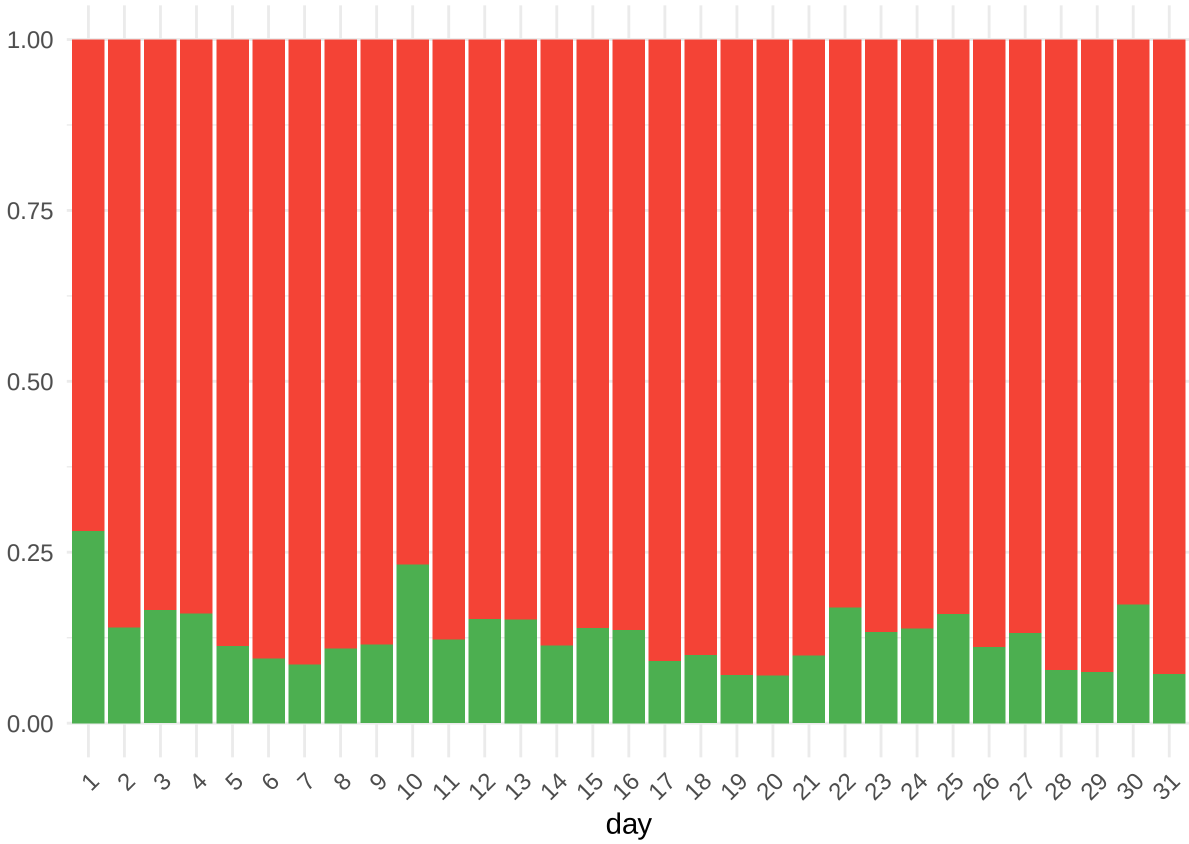
```
create_bar_plot(bank_data, "contact_type")
```



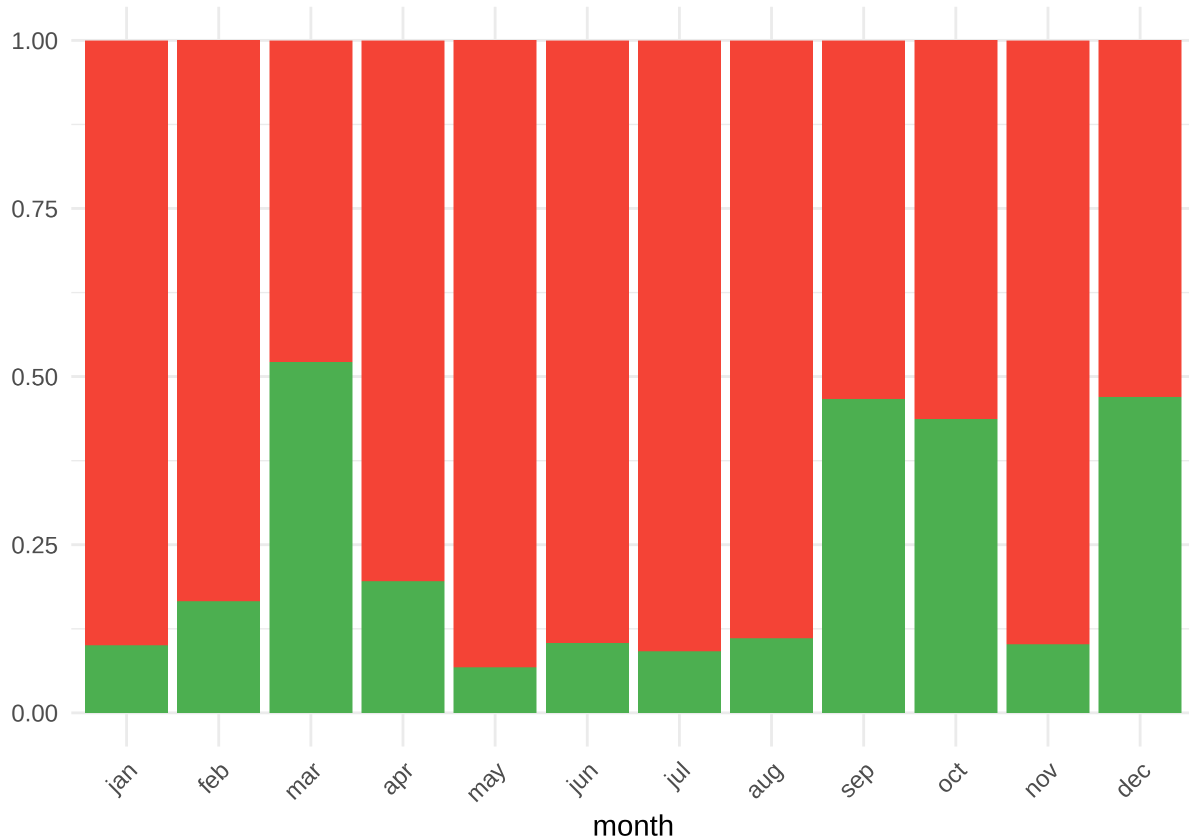
1.00
0.75
0.50
0.25
0.00
unknown
cellular
telephone
contact_type

5.10 Day and month

```r
create_bar_plot(bank_data, "day")
```

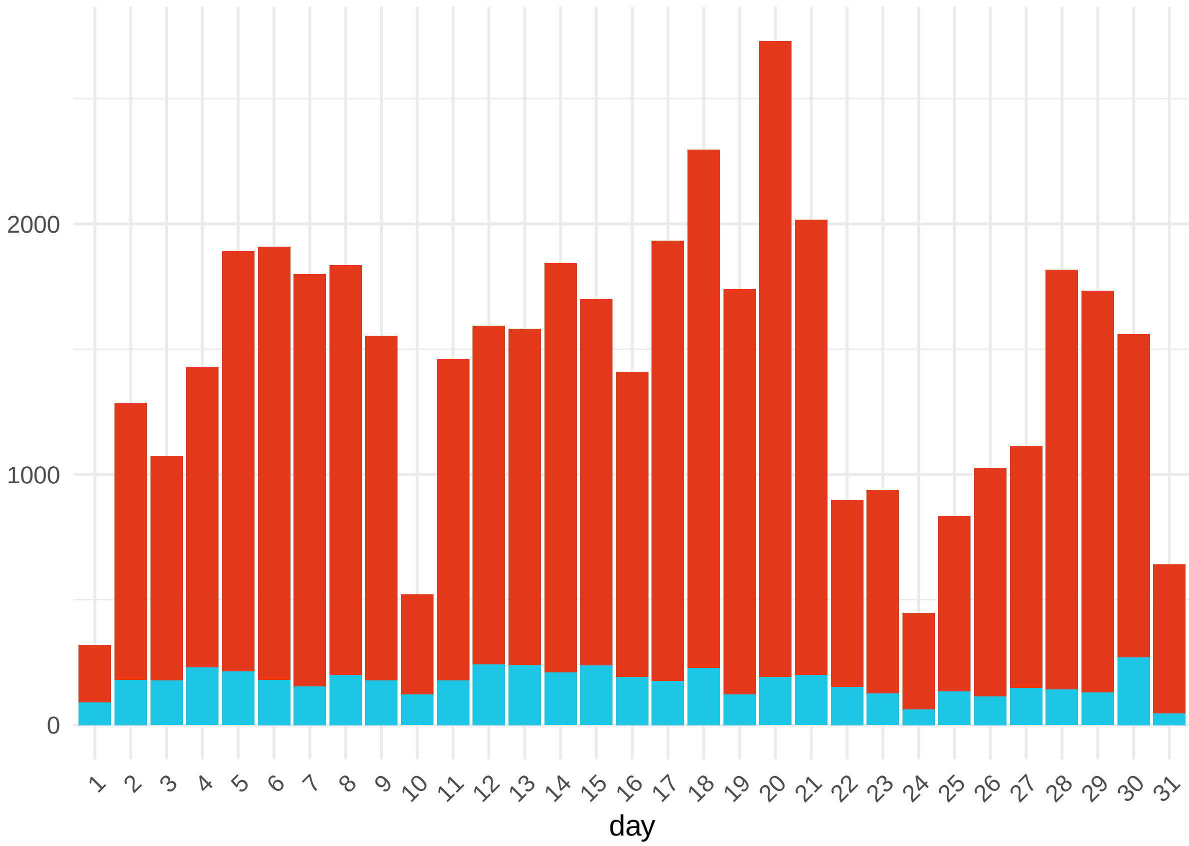



```r
create_bar_plot(bank_data, "month")
```

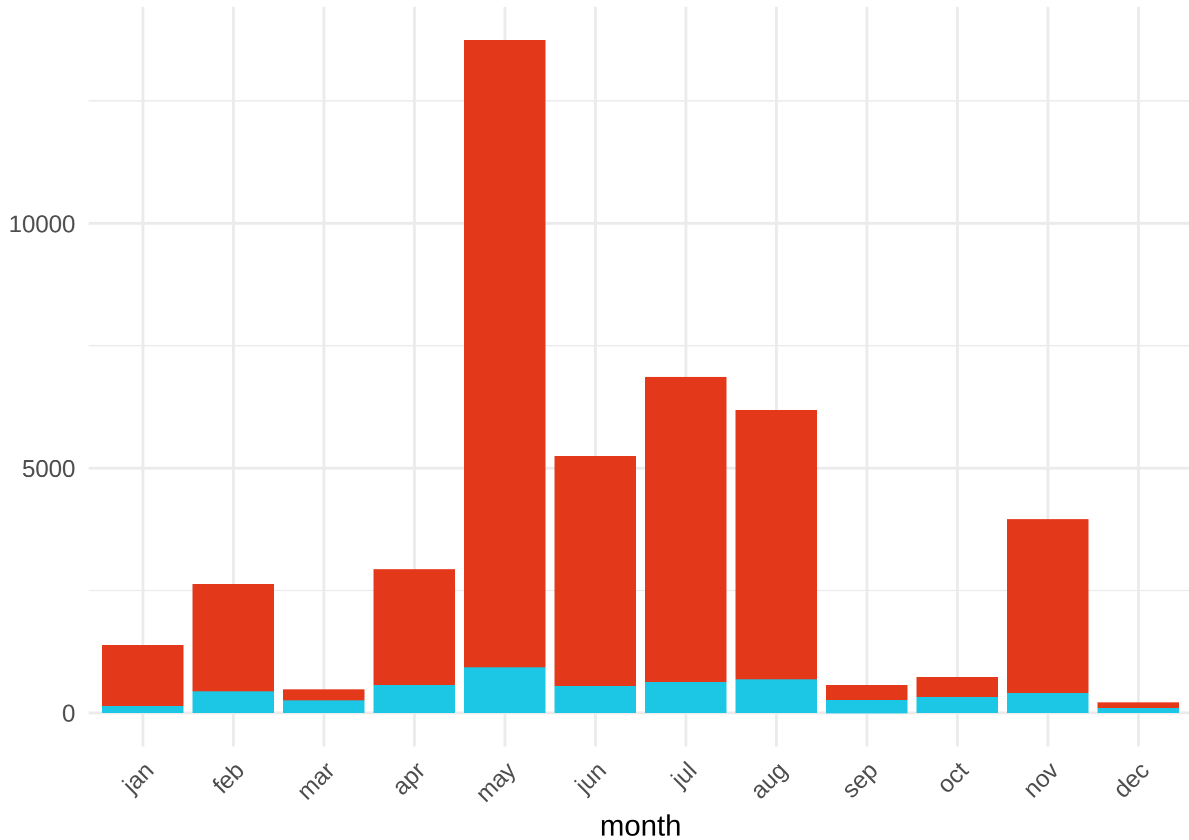


At first sight March, September, October and December seem to be the best months to contact the clients. Higher success could also be achieved when contacting the clients on the 1st, 10th, 22nd and 30th.

```
create_count_plot(bank_data, "day")
```



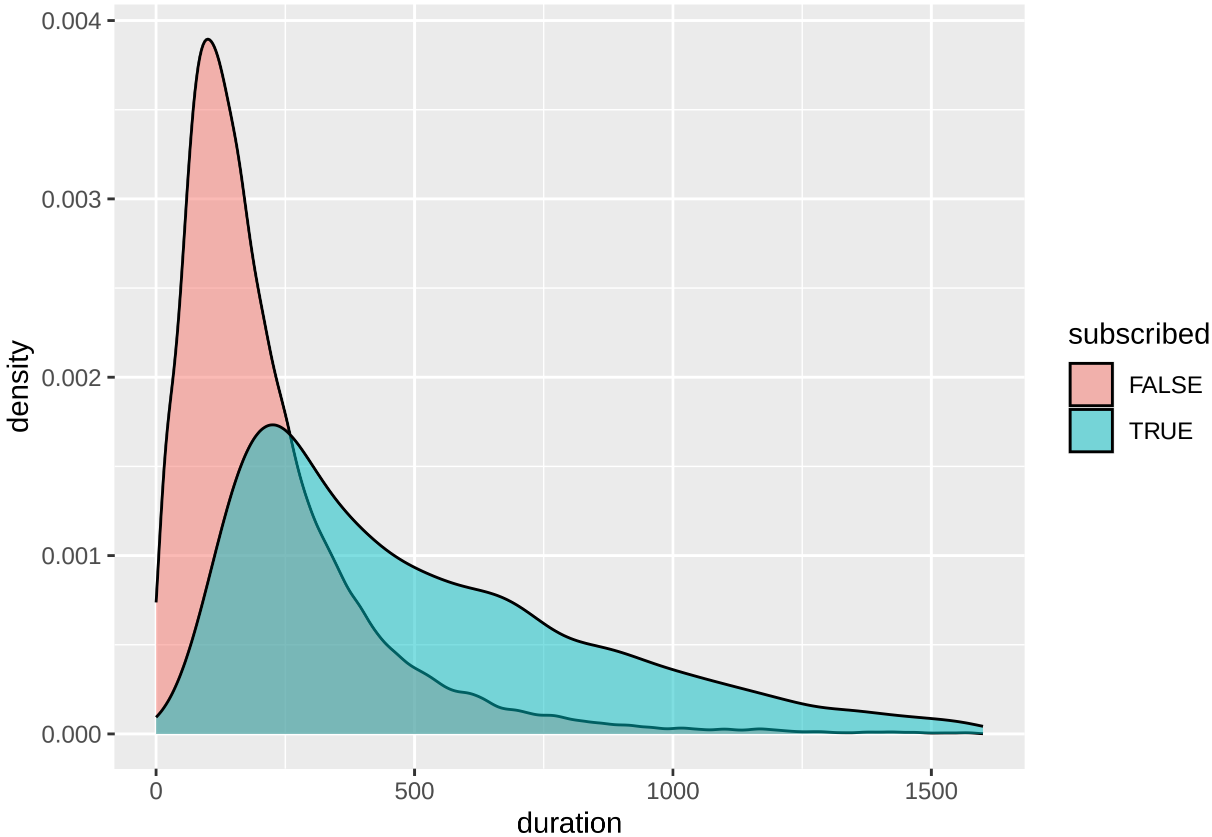

```
create_count_plot(bank_data, "month")
```



After digging deeper, it becomes clear that the distribution of number of calls by date and month is a disproportionate. The least clients were called on the 1st, 10th, 24th and 31st, and in March, September, October and December. This inconsistency should be addressed by the researchers that collected the data: more calls should be conducted to equalize the distribution.

5.11 Duration

```
ggplot(bank_data, 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. Yet, we will not be able to use this variable as it appears only after a call has taken place (we are trying to pick which clients to call).

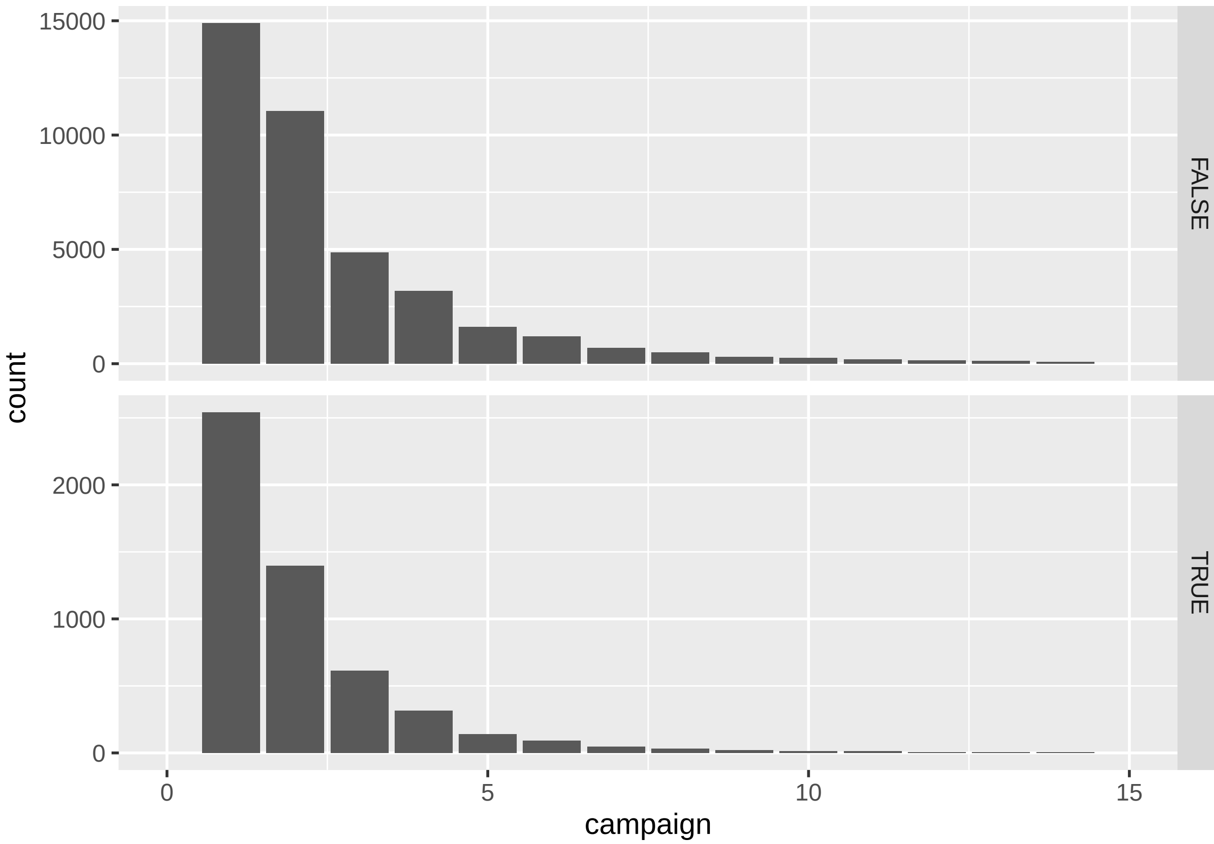
5.12 Attributes related to previous contact

5.12.1 Campaign contacts

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

```
## 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()`).
```




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

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

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

```
subscribed_camp <- bank_data$subscribed[bank_data$campaign < 6]
campaign_camp <- bank_data$campaign[bank_data$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.

5.12.2 Previous days

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

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

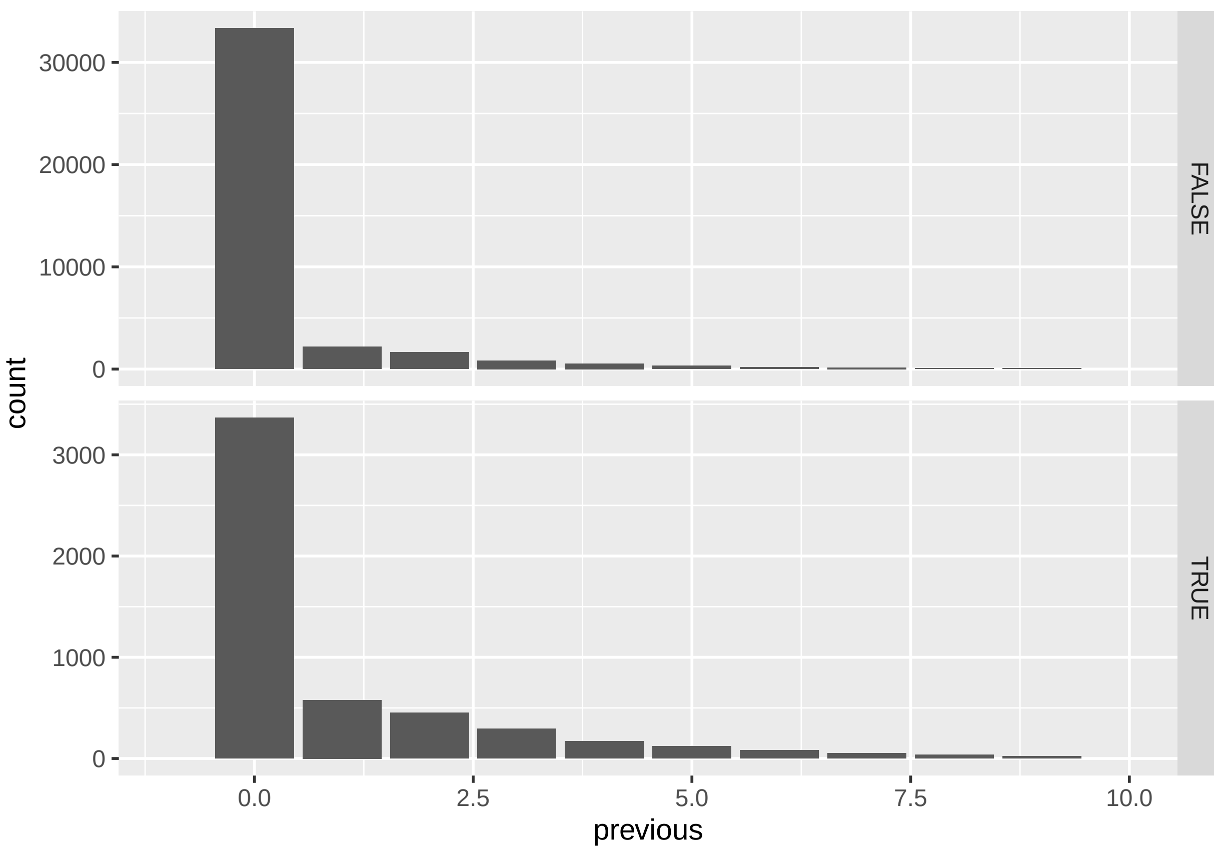
There are 8224 clients which have been contacted in the past. Since there are many different pdays values and because the variable has been encoded as -1 or any other natural number, we can transform this variable in to a binary variable.

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

5.12.3 Previous contacts

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

```
## 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()`).
```




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

```r
subscribed_prev <- bank_data$subscribed[bank_data$previous < 7]
previous_prev <- bank_data$previous[bank_data$previous < 7]

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

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44138
##
##
##               | subscribed_prev
## previous_prev |     FALSE |      TRUE | Row Total |
## --------------|-----------|-----------|-----------|
##             0 |     33333 |      3366 |     36699 |
##               |     0.908 |     0.092 |     0.831 |
##               |     0.853 |     0.665 |           |
## --------------|-----------|-----------|-----------|
##             1 |      2184 |       578 |      2762 |
##               |     0.791 |     0.209 |     0.063 |
##               |     0.056 |     0.114 |           |
## --------------|-----------|-----------|-----------|
##             2 |      1645 |       451 |      2096 |
##               |     0.785 |     0.215 |     0.047 |
##               |     0.042 |     0.089 |           |
## --------------|-----------|-----------|-----------|
##             3 |       847 |       292 |      1139 |
##               |     0.744 |     0.256 |     0.026 |
##               |     0.022 |     0.058 |           |
## --------------|-----------|-----------|-----------|
##             4 |       541 |       170 |       711 |
##               |     0.761 |     0.239 |     0.016 |
##               |     0.014 |     0.034 |           |
## --------------|-----------|-----------|-----------|
##             5 |       336 |       120 |       456 |
##               |     0.737 |     0.263 |     0.010 |
##               |     0.009 |     0.024 |           |
## --------------|-----------|-----------|-----------|
##             6 |       193 |        82 |       275 |
##               |     0.702 |     0.298 |     0.006 |
##               |     0.005 |     0.016 |           |
## --------------|-----------|-----------|-----------|
##  Column Total |     39079 |      5059 |     44138 |
##               |     0.885 |     0.115 |           |
## --------------|-----------|-----------|-----------|
##
##
```

Number of contacts during the previous campaign seems to linearly increase the likeliness of subscription. This is due to the client being interested in subscribing to a deposit.

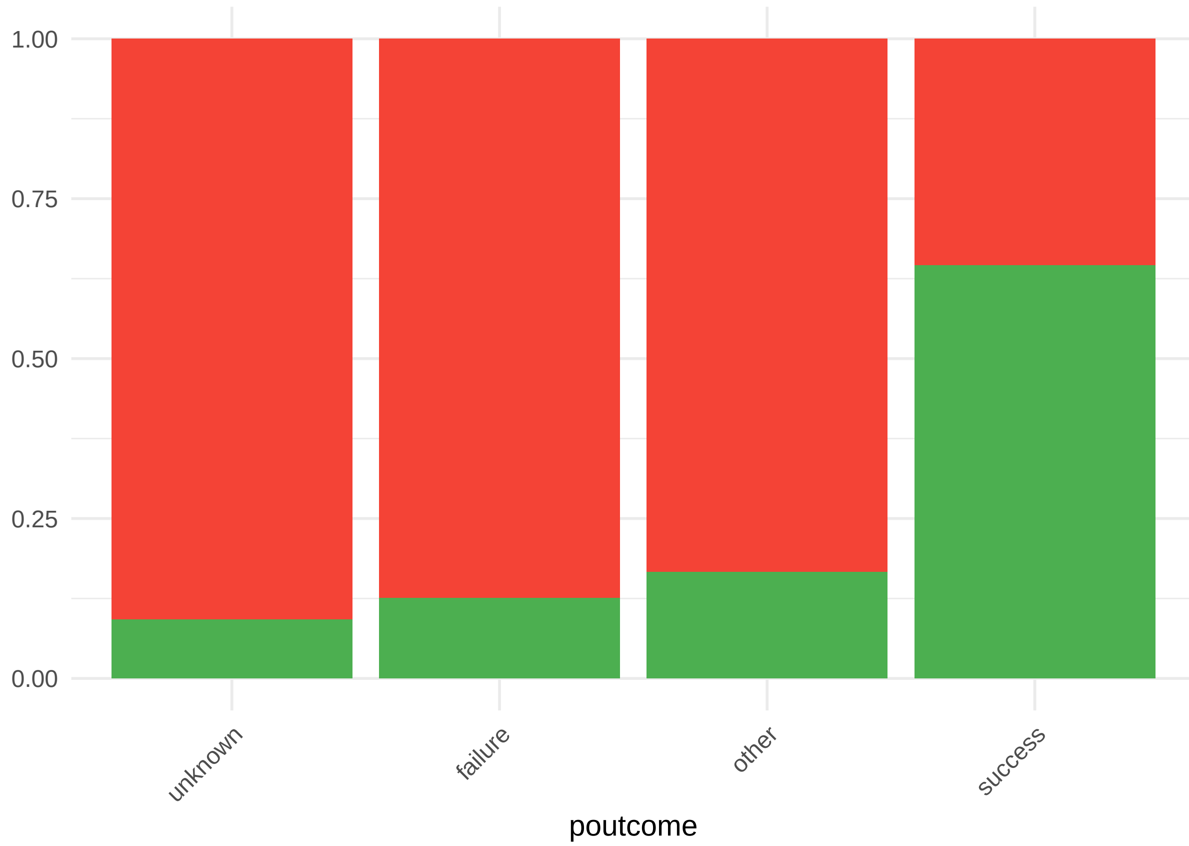
5.12.4 Previous outcome

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

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  44923
##
##
##                      | bank_data$poutcome
## bank_data$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 and the vast majority (81,7%) are set as unknown.

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


1.00
0.75
0.50
0.25
0.00
unknown
failure
other
success
poutcome

5.13 Correlation of continuous variables

```r
corr_matrix <- cor(bank_data[, 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.

6 Manipulating data (illustrative)

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_data, 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_data %>%
  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_data %>%
  filter((housing_loan == TRUE | personal_loan == TRUE) & age >= median(age, na.rm = TRUE))
```

We may also calculate the summarizing statistics.

```
job_summary <- bank_data %>%
  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.

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

```
## Registered S3 method overwritten by 'DescTools':
##   method         from
##   reorder.factor gdata
```

```
print(subscriber_summary)
```

```
##   mode_age   mode_job mode_marital mode_education mode_balance
## 1       32 management      married      secondary            0
##   mode_housing_loan mode_personal_loan mode_contact_type mode_day mode_month
## 1             FALSE              FALSE          cellular       30        may
##   mode_duration mode_campaign mode_pdays mode_previous mode_poutcome
## 1           261             1         -1             0       unknown
##   mode_subscribed mode_age_categ mode_trans_balance mode_was_contacted
## 1            TRUE            mid          -1.162368              FALSE
```

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.

```
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_data <- bank_data %>%
  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.

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

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

We can see that to 5227 “potential” clients the marketing campaign hasn’t been effective.

7 Modelling

Our original dataset is separated into two, training and testing, inside the create_model_data function.

```
dummy_full <- create_model_data(bank_data)
```

And finally, we can run the model and output the results.

```
results_1 <- train_evaluate_model(dummy_full, "Full Model")
```

```
## Setting levels: control = FALSE, case = TRUE
```

```
## Setting direction: controls < cases
```

```
cat("Model 1 (Full) Summary:\n")
```

```
## Model 1 (Full) Summary:
```

```
print(summary(results_1$model))
```

```
##
## Call:
## glm(formula = subscribed ~ ., family = binomial, data = train_data)
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -2.654e+00  2.685e-01  -9.888  < 2e-16 ***
## age_categ.high          3.753e-01  1.411e-01   2.660 0.007808 **
## age_categ.mid          -5.795e-01  9.429e-02  -6.146 7.94e-10 ***
## was_contactedTRUE       9.621e-01  1.053e+00   0.914 0.360977
## job.admin              -3.092e-02  1.119e-01  -0.276 0.782358
## job.blue.collar        -1.633e-01  1.104e-01  -1.479 0.139082
## job.entrepreneur       -1.721e-01  1.487e-01  -1.157 0.247148
## job.housemaid          -3.981e-01  1.571e-01  -2.534 0.011262 *
## job.management         -4.090e-02  1.092e-01  -0.375 0.707946
## job.retired            -3.137e-02  1.334e-01  -0.235 0.814070
## job.self.employed      -6.591e-02  1.376e-01  -0.479 0.631892
## job.services           -1.259e-01  1.191e-01  -1.057 0.290365
## job.student             1.180e-01  1.430e-01   0.826 0.409065
## job.technician         -9.385e-02  1.086e-01  -0.864 0.387487
## marital.divorced       -9.183e-02  6.321e-02  -1.453 0.146291
## marital.married        -2.782e-01  4.290e-02  -6.484 8.93e-11 ***
## education.primary      -1.143e-01  1.070e-01  -1.068 0.285405
## education.secondary     9.567e-03  9.524e-02   0.100 0.919983
## education.tertiary      1.272e-01  9.949e-02   1.279 0.200914
## contact_type.cellular   1.256e+00  7.232e-02  17.371  < 2e-16 ***
## contact_type.telephone  9.328e-01  9.977e-02   9.349  < 2e-16 ***
## day.2                  -2.274e-01  1.888e-01  -1.205 0.228369
## day.3                  -8.694e-03  1.900e-01  -0.046 0.963505
## day.4                  -7.773e-02  1.831e-01  -0.424 0.671224
## day.5                  -2.095e-01  1.841e-01  -1.138 0.255249
## day.6                  -6.901e-02  1.837e-01  -0.376 0.707198
## day.7                  -3.070e-01  1.895e-01  -1.620 0.105187
## day.8                   1.290e-02  1.829e-01   0.070 0.943806
## day.9                  -8.021e-02  1.892e-01  -0.424 0.671594
## day.10                  4.120e-01  2.088e-01   1.973 0.048468 *
## day.11                 -5.308e-02  1.866e-01  -0.284 0.776092
```

```
## day.12                2.408e-01  1.829e-01   1.317 0.187919
## day.13                3.642e-01  1.831e-01   1.989 0.046723 *
## day.14                9.077e-02  1.840e-01   0.493 0.621751
## day.15                1.581e-01  1.829e-01   0.864 0.387368
## day.16                1.046e-01  1.859e-01   0.563 0.573684
## day.17               -3.483e-01  1.845e-01  -1.888 0.059050 .
## day.18               -1.327e-01  1.804e-01  -0.736 0.462028
## day.19               -4.223e-01  1.969e-01  -2.145 0.031976 *
## day.20               -4.908e-01  1.841e-01  -2.666 0.007682 **
## day.21               -1.544e-02  1.862e-01  -0.083 0.933908
## day.22                1.393e-01  1.960e-01   0.711 0.477313
## day.23                3.911e-01  2.020e-01   1.936 0.052844 .
## day.24                1.539e-02  2.300e-01   0.067 0.946649
## day.25                2.312e-01  1.978e-01   1.169 0.242269
## day.26                2.062e-02  2.037e-01   0.101 0.919348
## day.27                2.725e-01  1.965e-01   1.387 0.165590
## day.28               -1.608e-01  1.962e-01  -0.820 0.412426
## day.29               -2.203e-01  1.985e-01  -1.110 0.267089
## day.30                4.101e-01  1.829e-01   2.242 0.024947 *
## day.31               -1.779e-01  2.450e-01  -0.726 0.467776
## month.feb             7.591e-01  1.421e-01   5.343 9.14e-08 ***
## month.mar             2.048e+00  1.648e-01  12.430  < 2e-16 ***
## month.apr             1.105e+00  1.335e-01   8.277  < 2e-16 ***
## month.may             4.936e-01  1.312e-01   3.763 0.000168 ***
## month.jun             1.326e+00  1.436e-01   9.240  < 2e-16 ***
## month.jul             3.415e-01  1.302e-01   2.622 0.008740 **
## month.aug             2.757e-01  1.314e-01   2.098 0.035932 *
## month.sep             1.450e+00  1.615e-01   8.976  < 2e-16 ***
## month.oct             1.617e+00  1.517e-01  10.660  < 2e-16 ***
## month.nov             4.435e-01  1.419e-01   3.126 0.001772 **
## month.dec             1.484e+00  2.100e-01   7.066 1.59e-12 ***
## poutcome.failure     -1.012e+00  1.051e+00  -0.964 0.335196
## poutcome.other       -7.449e-01  1.052e+00  -0.708 0.479001
## poutcome.success      1.101e+00  1.052e+00   1.047 0.295173
## balance               1.443e-05  5.161e-06   2.796 0.005171 **
## campaign             -7.420e-02  9.489e-03  -7.820 5.29e-15 ***
## pdays                -9.200e-05  3.118e-04  -0.295 0.767912
## previous              9.688e-03  6.232e-03   1.555 0.120061
## in_defaultTRUE        6.290e-02  1.565e-01   0.402 0.687807
## housing_loanTRUE     -4.388e-01  4.339e-02 -10.112  < 2e-16 ***
## personal_loanTRUE    -3.597e-01  5.944e-02  -6.051 1.44e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25938  on 35938  degrees of freedom
## Residual deviance: 21424  on 35867  degrees of freedom
## AIC: 21568
##
## Number of Fisher Scoring iterations: 6
```

```r
cat("\nModel 1 AUC:", round(results_1$auc, 4), "\n")
```

```
##
```

```
## Model 1 AUC: 0.7839
```

Now we remove variables that are not statistically meaningful to the model and produce a second much simpler model.

```r
exclude_vars <- c(
  "job.admin", "job.blue.collar", "job.entrepreneur", "job.management",
  "job.retired", "job.self.employed", "job.services", "job.technician",
  "marital.divorced", "education.secondary",
  paste0("day.", c(2:6, 7:9, 11:12, 14:16, 18, 20:22, 24:26, 28:29, 31)),
  "campaign", "poutcome.failure", "poutcome.other", "poutcome.success",
  "in_default", "previous", "balance"
)

dummy_full_2 <- create_model_data(bank_data, exclude_vars = exclude_vars)
results_2 <- train_evaluate_model(dummy_full_2, "Reduced Model")
```

```
## Setting levels: control = FALSE, case = TRUE
## Setting direction: controls < cases
```

```r
cat("\nModel 2 (Reduced) Summary:\n")
```

```
##
## Model 2 (Reduced) Summary:
```

```r
print(summary(results_2$model))
```

```
##
## Call:
## glm(formula = subscribed ~ ., family = binomial, data = train_data)
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -3.0189821  0.1577062 -19.143  < 2e-16 ***
## age_categ.high         0.4790406  0.1198960   3.995 6.46e-05 ***
## age_categ.mid         -0.6263293  0.0911891  -6.868 6.49e-12 ***
## was_contactedTRUE      1.0770417  0.0723137  14.894  < 2e-16 ***
## job.housemaid         -0.3227343  0.1214678  -2.657 0.007885 **
## job.student            0.2187663  0.1020065   2.145 0.031982 *
## marital.married       -0.2542264  0.0370561  -6.861 6.86e-12 ***
## education.primary     -0.1783262  0.0588237  -3.032 0.002433 **
## education.tertiary     0.1624708  0.0390354   4.162 3.15e-05 ***
## contact_type.cellular  1.2885364  0.0708654  18.183  < 2e-16 ***
## contact_type.telephone 0.9042723  0.0973182   9.292  < 2e-16 ***
## day.10                 0.6422168  0.1307393   4.912 9.01e-07 ***
## day.13                 0.4572996  0.0872949   5.239 1.62e-07 ***
## day.17                -0.2632849  0.0945679  -2.784 0.005368 **
## day.19                -0.3495810  0.1144201  -3.055 0.002249 **
## day.23                 0.4747969  0.1242516   3.821 0.000133 ***
## day.27                 0.3295593  0.1123365   2.934 0.003350 **
## day.30                 0.4969487  0.0894006   5.559 2.72e-08 ***
## month.feb              0.7860672  0.1217950   6.454 1.09e-10 ***
## month.mar              2.1431394  0.1510740  14.186  < 2e-16 ***
## month.apr              1.1257197  0.1188761   9.470  < 2e-16 ***
## month.may              0.6337131  0.1164408   5.442 5.26e-08 ***
## month.jun              1.4269731  0.1265940  11.272  < 2e-16 ***
```

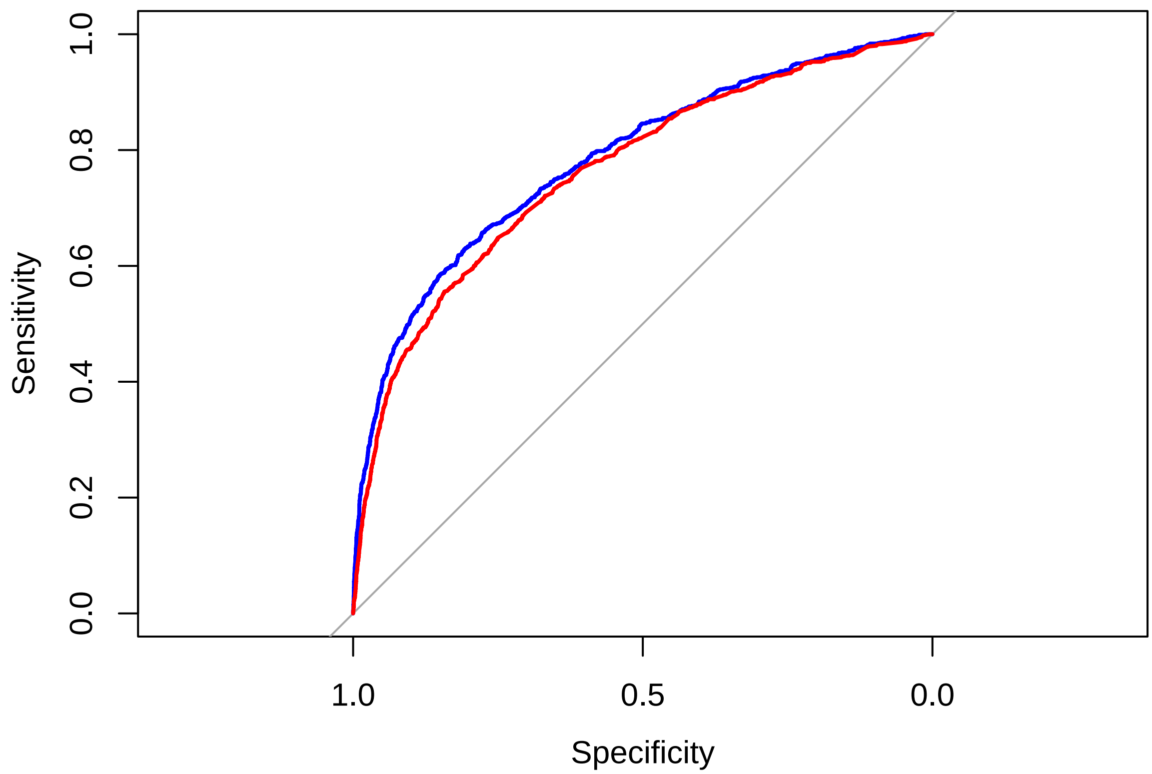
```
## month.jul              0.4429159  0.1174891   3.770 0.000163 ***
## month.aug              0.3297298  0.1171340   2.815 0.004878 **
## month.sep              1.7143563  0.1460299  11.740  < 2e-16 ***
## month.oct              1.8010611  0.1392273  12.936  < 2e-16 ***
## month.nov              0.4195440  0.1236837   3.392 0.000694 ***
## month.dec              1.7219653  0.1926659   8.938  < 2e-16 ***
## pdays                 -0.0020345  0.0002932  -6.939 3.95e-12 ***
## housing_loanTRUE      -0.5551318  0.0416664 -13.323  < 2e-16 ***
## personal_loanTRUE     -0.4342465  0.0580883  -7.476 7.68e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25938  on 35938  degrees of freedom
## Residual deviance: 22334  on 35907  degrees of freedom
## AIC: 22398
##
## Number of Fisher Scoring iterations: 6
```

```r
cat("\nModel 2 AUC:", round(results_2$auc, 4), "\n")
```

```
##
## Model 2 AUC: 0.7652
```

We should also compare the models' performance.

```r
plot(results_1$roc_curve, col = "blue")
plot(results_2$roc_curve, col = "red", add = TRUE)
```



Although, with the statistically insignificant parameters removed, our logistic regression model's AUC is lowered to 0,7652 from 0,7839, the model becomes much simpler which is prefered.

```r
pred_class <- ifelse(results_2$predictions > 0.5, TRUE, FALSE)
confusion_matrix <- confusionMatrix(
  factor(pred_class),
  factor(results_2$test_data$subscribed),
  positive = "TRUE"
)
print(confusion_matrix)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE TRUE
##      FALSE  7845  934
##      TRUE     88  117
##
##                Accuracy : 0.8862
##                  95% CI : (0.8795, 0.8927)
##     No Information Rate : 0.883
##     P-Value [Acc > NIR] : 0.1749
##
##                   Kappa : 0.154
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.11132
##             Specificity : 0.98891
##          Pos Pred Value : 0.57073
##          Neg Pred Value : 0.89361
##              Prevalence : 0.11699
##          Detection Rate : 0.01302
##    Detection Prevalence : 0.02282
##       Balanced Accuracy : 0.55011
##
##        'Positive' Class : TRUE
##
```

The sensitivity (true positive) of the model is quite low. Only 11,8% of clients who would subscribe to a deposit are being recognized as "subscribers".

We can try lowering the threshold as there is no need to be too conservative.

```r
pred_class_2 <- ifelse(results_2$predictions > 0.12, TRUE, FALSE)
confusion_matrix_2 <- confusionMatrix(
  factor(pred_class_2),
  factor(results_2$test_data$subscribed),
  positive = "TRUE"
)
print(confusion_matrix_2)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE TRUE
##      FALSE  6023  383
```

```
##      TRUE   1910  668
##
##                Accuracy : 0.7448
##                  95% CI : (0.7356, 0.7538)
##     No Information Rate : 0.883
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2422
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.63559
##             Specificity : 0.75923
##          Pos Pred Value : 0.25912
##          Neg Pred Value : 0.94021
##              Prevalence : 0.11699
##          Detection Rate : 0.07435
##    Detection Prevalence : 0.28695
##       Balanced Accuracy : 0.69741
##
##        'Positive' Class : TRUE
##
```

By lowering the threshold down to 0.12, true positives are being recognized with 63,56% accuracy (up from 11,13%) and the specificity is lowered to 75,92% (from 97,4%).

8 Conclusion

1. The logistic regression model accuracy score is 0,7448 (with threshold adjusted). True positive rate is 63,56%.
2. Most important parameters for choosing a potential bank deposit subscriber are contact type, day and month of contact and whether or not the client has borrowed a loan.
3. The duration variable cannot be used in the model due to in not being available before calling the customer even though it is the most significant determining feature.
4. The researchers should look into gathering additional data during aforementioned days and months.