Term Deposit Subscription Prediction Analysis

Introduction

The data is related with direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, in order to access if the product (bank term deposit) would be ('yes') or not ('no') subscribed.

```
require(ggplot2)
require(dplyr)
require(Hmisc)
require(reshape)
require(dummies)
require(caret)
require(ROCR)
require(randomForest)
require(caTools)
require(rpart)
require(rpart.plot)
```

Dataset

```
setwd('/Users/serhansuer/Desktop')
data <- read.csv('bank-full.csv',sep=';')
```

```
dim(data)
```

```
## [1] 45211    17
```

```
head(data)
```

```
##   age          job marital education default balance housing loan contact
## 1  58   management married  tertiary      no    2143     yes   no unknown
## 2  44   technician  single secondary      no      29     yes   no unknown
## 3  33 entrepreneur married secondary      no       2     yes  yes unknown
## 4  47  blue-collar married   unknown      no    1506     yes   no unknown
## 5  33      unknown  single   unknown      no       1      no   no unknown
## 6  35   management married  tertiary      no     231     yes   no unknown
##   day month duration campaign pdays previous poutcome  y
## 1   5   may      261        1    -1        0  unknown no
## 2   5   may      151        1    -1        0  unknown no
## 3   5   may       76        1    -1        0  unknown no
## 4   5   may       92        1    -1        0  unknown no
## 5   5   may      198        1    -1        0  unknown no
## 6   5   may      139        1    -1        0  unknown no
```

1. age (numeric)

2. job : type of job (categorical: ‘admin.’,‘blue-collar’,‘entrepreneur’,‘housemaid’,‘management’,‘retired’,‘self-employed’,‘services’,‘student’,‘technician’,‘unemployed’,‘unknown’)
3. marital : marital status (categorical: ‘divorced’,‘married’,‘single’,‘unknown’; note: ‘divorced’ means divorced or widowed)
4. education (categorical: ‘basic.4y’,‘basic.6y’,‘basic.9y’,‘high.school’,‘illiterate’,‘professional.course’,‘university.degree’,‘unknown’)
5. default: has credit in default? (categorical: ‘no’,‘yes’,‘unknown’)
6. balance: amount of money in customer’s account (numeric)
7. housing: has housing loan? (categorical: ‘no’,‘yes’,‘unknown’)
8. loan: has personal loan? (categorical: ‘no’,‘yes’,‘unknown’)
9. contact: contact communication type (categorical: ‘cellular’,‘telephone’)
10. month: last contact month of year (categorical: ‘jan’, ‘feb’, ‘mar’, …, ‘nov’, ‘dec’)
11. day_of_week: last contact day of the week (categorical: ‘mon’,‘tue’,‘wed’,‘thu’,‘fri’)
12. duration: last contact duration, in seconds (numeric).
13. campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)
14. pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; -1 means client was not previously contacted)
15. previous: number of contacts performed before this campaign and for this client (numeric)
16. poutcome: outcome of the previous marketing campaign (categorical: ‘failure’,‘nonexistent’,‘success’)
17. y - has the client subscribed a term deposit? (binary: ‘yes’,‘no’)

Exploratory Data Analysis

```
glimpse(data)
```

```
## Observations: 45,211
## Variables: 17
## $ age       <int> 58, 44, 33, 47, 33, 35, 28, 42, 58, 43, 41, 29, 53, 58…
## $ job       <fct> management, technician, entrepreneur, blue-collar, unk…
## $ marital   <fct> married, single, married, married, single, married, si…
## $ education <fct> tertiary, secondary, secondary, unknown, unknown, tert…
## $ default   <fct> no, no, no, no, no, no, no, yes, no, no, no, no, no, n…
## $ balance   <int> 2143, 29, 2, 1506, 1, 231, 447, 2, 121, 593, 270, 390,…
## $ housing   <fct> yes, yes, yes, yes, no, yes, yes, yes, yes, yes, yes, …
## $ loan      <fct> no, no, yes, no, no, no, yes, no, no, no, no, no, no, …
## $ contact   <fct> unknown, unknown, unknown, unknown, unknown, unknown, …
## $ day       <int> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, …
## $ month     <fct> may, may, may, may, may, may, may, may, may, may, may,…
## $ duration  <int> 261, 151, 76, 92, 198, 139, 217, 380, 50, 55, 222, 137…
## $ campaign  <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, …
## $ pdays     <int> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1…
## $ previous  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …
## $ poutcome  <fct> unknown, unknown, unknown, unknown, unknown, unknown, …
## $ y         <fct> no, no, no, no, no, no, no, no, no, no, no, no, no, no…
```

```
describe(data)
```

```
## data
##
##  17  Variables      45211  Observations
## ---------------------------------------------------------------------------
## age
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##    45211        0       77    0.999    40.94    11.87       27       29
##      .25      .50      .75      .90      .95
##       33       39       48       56       59
##
## lowest : 18 19 20 21 22, highest: 90 92 93 94 95
## ---------------------------------------------------------------------------
## job
##        n  missing distinct
##    45211        0       12
##
## admin. (5171, 0.114), blue-collar (9732, 0.215), entrepreneur (1487,
## 0.033), housemaid (1240, 0.027), management (9458, 0.209), retired (2264,
## 0.050), self-employed (1579, 0.035), services (4154, 0.092), student (938,
## 0.021), technician (7597, 0.168), unemployed (1303, 0.029), unknown (288,
## 0.006)
## ---------------------------------------------------------------------------
## marital
##        n  missing distinct

##    45211        0        3
##
## Value      divorced  married   single
## Frequency      5207    27214    12790
## Proportion    0.115    0.602    0.283
## ---------------------------------------------------------------------------
## education
##        n  missing distinct
##    45211        0        4
##
## Value        primary secondary  tertiary   unknown
## Frequency       6851     23202     13301      1857
## Proportion     0.152     0.513     0.294     0.041
## ---------------------------------------------------------------------------
## default
##        n  missing distinct
##    45211        0        2
##
## Value         no   yes
## Frequency  44396   815
## Proportion 0.982 0.018
## ---------------------------------------------------------------------------
## balance
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##    45211        0     7168        1     1362     2054     -172        0
##      .25      .50      .75      .90      .95
##       72      448     1428     3574     5768
##
## lowest :  -8019  -6847  -4057  -3372  -3313, highest:  66721  71188  81204  98417
102127
## ---------------------------------------------------------------------------
## housing
##        n  missing distinct
```

```
##        n  missing distinct
##    45211        0        2
##
## Value         no   yes
## Frequency  20081 25130
## Proportion 0.444 0.556
## ---------------------------------------------------------------------------
## loan
##        n  missing distinct
##    45211        0        2
##
## Value         no   yes
## Frequency  37967  7244
## Proportion  0.84  0.16
## ---------------------------------------------------------------------------
## contact
##        n  missing distinct
##    45211        0        3
##
## Value       cellular telephone   unknown
## Frequency      29285      2906     13020
## Proportion     0.648     0.064     0.288
## ---------------------------------------------------------------------------
## day
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##    45211        0       31    0.999    15.81    9.576        3        5
##      .25      .50      .75      .90      .95
##        8       16       21       28       29
##
## lowest :  1  2  3  4  5, highest: 27 28 29 30 31
## ---------------------------------------------------------------------------
## month
##        n  missing distinct
##    45211        0       12
##
## Value        apr   aug   dec   feb   jan   jul   jun   mar   may   nov
## Frequency   2932  6247   214  2649  1403  6895  5341   477 13766  3970
## Proportion 0.065 0.138 0.005 0.059 0.031 0.153 0.118 0.011 0.304 0.088
##
## Value        oct   sep
## Frequency    738   579
## Proportion 0.016 0.013
## ---------------------------------------------------------------------------
## duration
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##    45211        0     1573        1    258.2    235.4       35       58
##      .25      .50      .75      .90      .95
##      103      180      319      548      751
##
## lowest :    0    1    2    3    4, highest: 3366 3422 3785 3881 4918
## ---------------------------------------------------------------------------
## campaign
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##    45211        0       48    0.918    2.764    2.383        1        1
##      .25      .50      .75      .90      .95
##        1        2        3        5        8
##
## lowest :  1  2  3  4  5, highest: 50 51 55 58 63
## ---------------------------------------------------------------------------
```

```
## pdays
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##    45211        0      559    0.454     40.2    71.61       -1       -1
##      .25      .50      .75      .90      .95
##       -1       -1       -1      185      317
##
## lowest :  -1   1   2   3   4, highest: 838 842 850 854 871
## ---------------------------------------------------------------------------
## previous
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##    45211        0       41    0.454   0.5803    1.044        0        0
##      .25      .50      .75      .90      .95
##        0        0        0        2        3
##
## lowest :   0   1   2   3   4, highest:  41  51  55  58 275
## ---------------------------------------------------------------------------
## poutcome
##        n  missing distinct
##    45211        0        4
##
## Value      failure   other success unknown
## Frequency     4901    1840    1511   36959
## Proportion   0.108   0.041   0.033   0.817
## ---------------------------------------------------------------------------
## y
##        n  missing distinct
##    45211        0        2
##
## Value         no   yes
## Frequency  39922  5289
## Proportion 0.883 0.117
## ---------------------------------------------------------------------------
```

```
str(data)
```

```
## 'data.frame':    45211 obs. of  17 variables:
##  $ age      : int  58 44 33 47 33 35 28 42 58 43 ...
##  $ job      : Factor w/ 12 levels "admin.","blue-collar",..: 5 10 3 2 12 5 5 3 6 1
0 ...
##  $ marital  : Factor w/ 3 levels "divorced","married",..: 2 3 2 2 3 2 3 1 2 3 ...
##  $ education: Factor w/ 4 levels "primary","secondary",..: 3 2 2 4 4 3 3 3 1 2 ...
##  $ default  : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 2 1 1 ...
##  $ balance  : int  2143 29 2 1506 1 231 447 2 121 593 ...
##  $ housing  : Factor w/ 2 levels "no","yes": 2 2 2 2 1 2 2 2 2 2 ...
##  $ loan     : Factor w/ 2 levels "no","yes": 1 1 2 1 1 1 2 1 1 1 ...
##  $ contact  : Factor w/ 3 levels "cellular","telephone",..: 3 3 3 3 3 3 3 3 3 3
...
##  $ day      : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ month    : Factor w/ 12 levels "apr","aug","dec",..: 9 9 9 9 9 9 9 9 9 9 ...
##  $ 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 "failure","other",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ y        : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
```

```
summary(data)
```

```
##       age                 job           marital          education    
##  Min.   :18.00   blue-collar:9732   divorced: 5207   primary  : 6851  
##  1st Qu.:33.00   management :9458   married :27214   secondary:23202  
##  Median :39.00   technician :7597   single  :12790   tertiary :13301  
##  Mean   :40.94   admin.     :5171                    unknown  : 1857  
##  3rd Qu.:48.00   services   :4154                                     
##  Max.   :95.00   retired    :2264                                     
##                  (Other)    :6835                                     
##  default        balance       housing      loan            contact     
##  no :44396   Min.   : -8019   no :20081   no :37967   cellular :29285  
##  yes:  815   1st Qu.:    72   yes:25130   yes: 7244   telephone: 2906  
##              Median :   448                           unknown  :13020  
##              Mean   :  1362                                            
##              3rd Qu.:  1428                                            
##              Max.   :102127                                            
##                                                                        
##       day            month          duration         campaign     
##  Min.   : 1.00   may    :13766   Min.   :   0.0   Min.   : 1.000  
##  1st Qu.: 8.00   jul    : 6895   1st Qu.: 103.0   1st Qu.: 1.000  
##  Median :16.00   aug    : 6247   Median : 180.0   Median : 2.000  
##  Mean   :15.81   jun    : 5341   Mean   : 258.2   Mean   : 2.764  
##  3rd Qu.:21.00   nov    : 3970   3rd Qu.: 319.0   3rd Qu.: 3.000  
##  Max.   :31.00   apr    : 2932   Max.   :4918.0   Max.   :63.000  
##                  (Other): 6060                                    
##      pdays          previous            poutcome       y        
##  Min.   : -1.0   Min.   :  0.0000   failure: 4901   no :39922  
##  1st Qu.: -1.0   1st Qu.:  0.0000   other  : 1840   yes: 5289  
##  Median : -1.0   Median :  0.0000   success: 1511              
##  Mean   : 40.2   Mean   :  0.5803   unknown:36959              
##  3rd Qu.: -1.0   3rd Qu.:  0.0000                              
##  Max.   :871.0   Max.   :275.0000                              
## 
```

When we check the summary table, we can say that there could be outliers in “campaign” (number of calls), “previous” (number of contacts in previous campaigns) and “duration” (of call) variables. Also column “pdays” (time passed after last call) has value of 999 in some rows meaning the customer has not received a call before. And since we have categorical variables, we will need to dummify them and scale them to numeric variables.

```
data_unq <- subset(data, select = -c(age, duration, balance, pdays))
unq_vals <- lapply(data_unq, unique)
unq_vals
```

```
## $job
##  [1] management    technician    entrepreneur  blue-collar   unknown
##  [6] retired       admin.        services      self-employed unemployed
## [11] housemaid     student
## 12 Levels: admin. blue-collar entrepreneur housemaid ... unknown
##
## $marital
## [1] married  single   divorced
## Levels: divorced married single
##
## $education
## [1] tertiary  secondary unknown   primary
## Levels: primary secondary tertiary unknown
##
## $default
## [1] no  yes
## Levels: no yes
##
## $housing
## [1] yes no
## Levels: no yes
##
## $loan
## [1] no  yes
## Levels: no yes
##
## $contact
## [1] unknown   cellular  telephone
## Levels: cellular telephone unknown
##
## $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
## [24] 17 18 24 25  1 10 22 31
##
## $month
##  [1] may jun jul aug oct nov dec jan feb mar apr sep
## Levels: apr aug dec feb jan jul jun mar may nov oct sep
##
## $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
## [24] 21 43 51 63 41 26 28 55 50 38 23 20 29 31 37 30 46 27 58 33 35 34 36
## [47] 39 44
##
## $previous
##  [1]   0   3   1   4   2  11  16   6   5  10  12   7  18   9  21   8  14
## [18]  15  26  37  13  25  20  27  17  23  38  29  24  51 275  22  19  30
## [35]  58  28  32  40  55  35  41
##
## $poutcome
## [1] unknown failure other   success
## Levels: failure other success unknown
##
## $y
## [1] no  yes
## Levels: no yes
```

```r
target <- 'y'
cat_vars <- c('job', 'marital', 'education', 'default', 'housing',
           'loan', 'contact', 'poutcome')

num_vars <- c('age', 'balance', 'day', 'duration', 'campaign', 'pdays', 'previous')
```

```r
for (i in cat_vars) {
    print(i)
    print(sort(table(data[i]), decreasing = TRUE))
    cat("\n")
}
```

```
## [1] "job"
##
##   blue-collar    management    technician        admin.      services
##          9732          9458          7597          5171          4154
##       retired self-employed  entrepreneur    unemployed     housemaid
##          2264          1579          1487          1303          1240
##       student       unknown
##           938           288
##
## [1] "marital"
##
##  married   single divorced
##    27214    12790     5207
##
## [1] "education"
##
## secondary  tertiary   primary   unknown
##     23202     13301      6851      1857
##
## [1] "default"
##
##    no   yes
## 44396   815
##
## [1] "housing"
##
##   yes    no
## 25130 20081
##
## [1] "loan"
##
##    no   yes
## 37967  7244
##
## [1] "contact"
##
##  cellular   unknown telephone
##     29285     13020      2906
##
## [1] "poutcome"
##
## unknown failure   other success
##   36959    4901    1840    1511
```

```
y_customers <- data %>%
  filter(y == "yes")
y_ratio <- nrow(y_customers) / nrow(data)
y_ratio
```

```
## [1] 0.1169848
```

Nearly 11.7% of our target value is yes which means 11.7% of all customers subscribed for term deposit.

```
monthly_results <- data %>%
    group_by(month) %>%
    summarise(yes=sum(y=="yes"), no= sum(y=="no"),perc=yes/(yes+no))%>%
    arrange(month)
monthly_results
```

```
## # A tibble: 12 x 4
##    month   yes    no   perc
##    <fct> <int> <int>  <dbl>
##  1 apr     577  2355 0.197
##  2 aug     688  5559 0.110
##  3 dec     100   114 0.467
##  4 feb     441  2208 0.166
##  5 jan     142  1261 0.101
##  6 jul     627  6268 0.0909
##  7 jun     546  4795 0.102
##  8 mar     248   229 0.520
##  9 may     925 12841 0.0672
## 10 nov     403  3567 0.102
## 11 oct     323   415 0.438
## 12 sep     269   310 0.465
```

It can be seen that month can affect the subscription result.

```
data%>%
  ggplot(aes(age))+
  geom_histogram(aes(fill=y),binwidth = 5)
```

Age distribution is positively skewed and when yes/no distributions checked amongst different ages, it looks like it might be a good predictor. Also, grouping ages according to life cycle changes like for example graduation, early professional years, later professional years, before retirement, after retirement might be useful.

```
data_job <-melt(data %>%
                mutate(rcount=1) %>%
                group_by(job,y) %>%
                summarise(sum(rcount)),id=c("job","y"))

jobsummary <- cast(data_job, job~y)
jobsummary %>%
    group_by(job) %>%
    mutate(percentage=round(yes/(yes+no),2)) %>%
    arrange(desc(percentage))
```

```
## # A tibble: 12 x 4
## # Groups:   job [12]
##    job              no   yes percentage
##    <fct>         <dbl> <dbl>      <dbl>
##  1 student         669   269      0.290
##  2 retired        1748   516      0.23
##  3 unemployed     1101   202      0.16
##  4 management     8157  1301      0.14
##  5 admin.         4540   631      0.12
##  6 self-employed  1392   187      0.12
##  7 unknown         254    34      0.12
##  8 technician     6757   840      0.11
##  9 housemaid      1131   109      0.09
## 10 services       3785   369      0.09
## 11 entrepreneur   1364   123      0.08
## 12 blue-collar    9024   708      0.07
```

When we grouped data by job and checked percentage of subscriptions top 3 is, student, retired and unemployed which means both of them are not currently employed, followed by admin. and management. This might give a clue about grouping job.

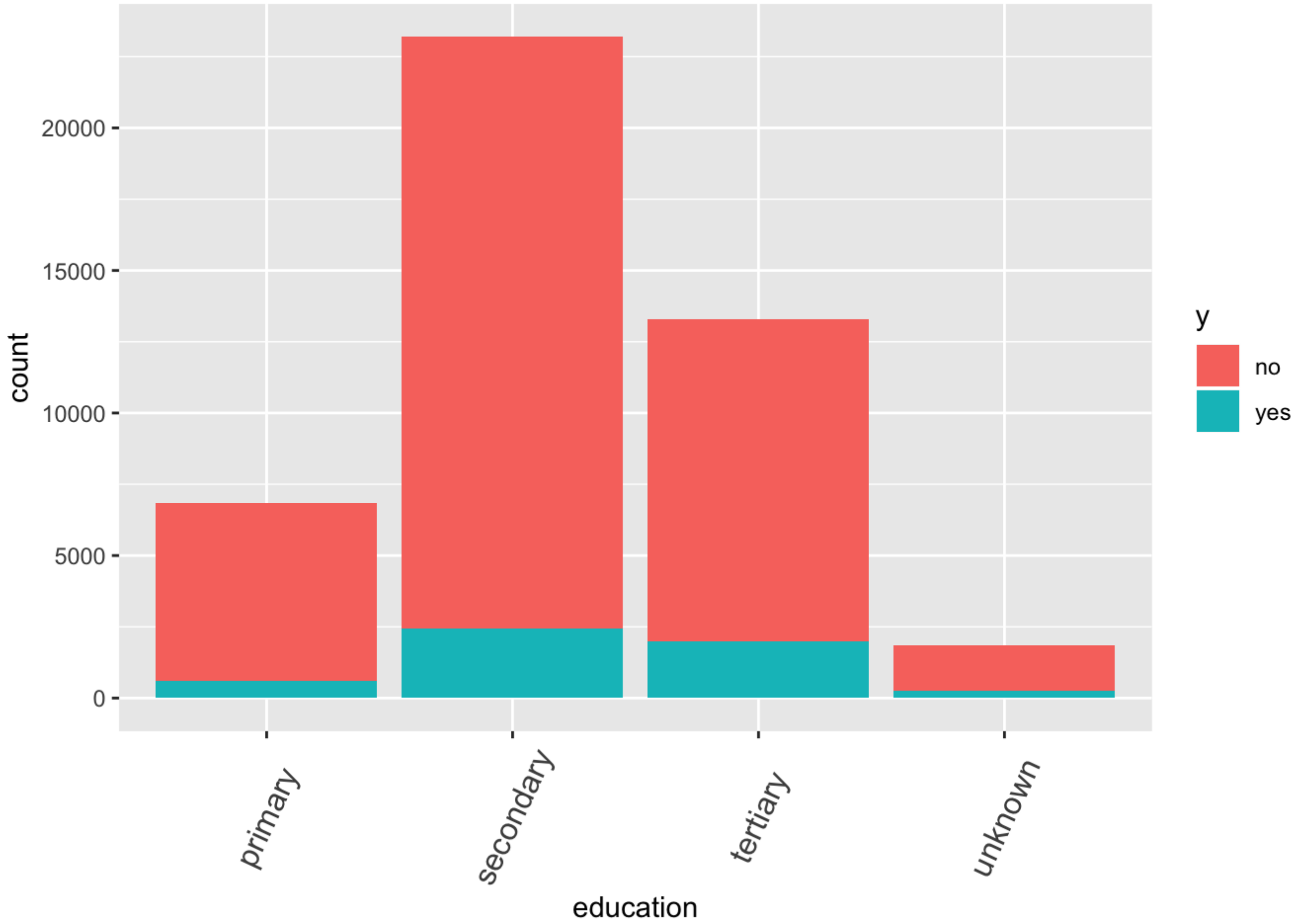
```
data_marital <- melt(data %>%
                        mutate(rcount=1) %>%
                        group_by(marital,y) %>%
                        summarise(sum(rcount)),id=c("marital","y"))

maritalsummary<- cast(data_marital, marital~y)
maritalsummary %>%
        group_by(marital) %>%
        mutate(percentage=round(yes/(yes+no),2)) %>%
        arrange(desc(percentage))
```

```
## # A tibble: 3 x 4
## # Groups:   marital [3]
##   marital     no   yes percentage
##   <fct>    <dbl> <dbl>      <dbl>
## 1 single   10878  1912       0.15
## 2 divorced  4585   622       0.12
## 3 married  24459  2755       0.1
```

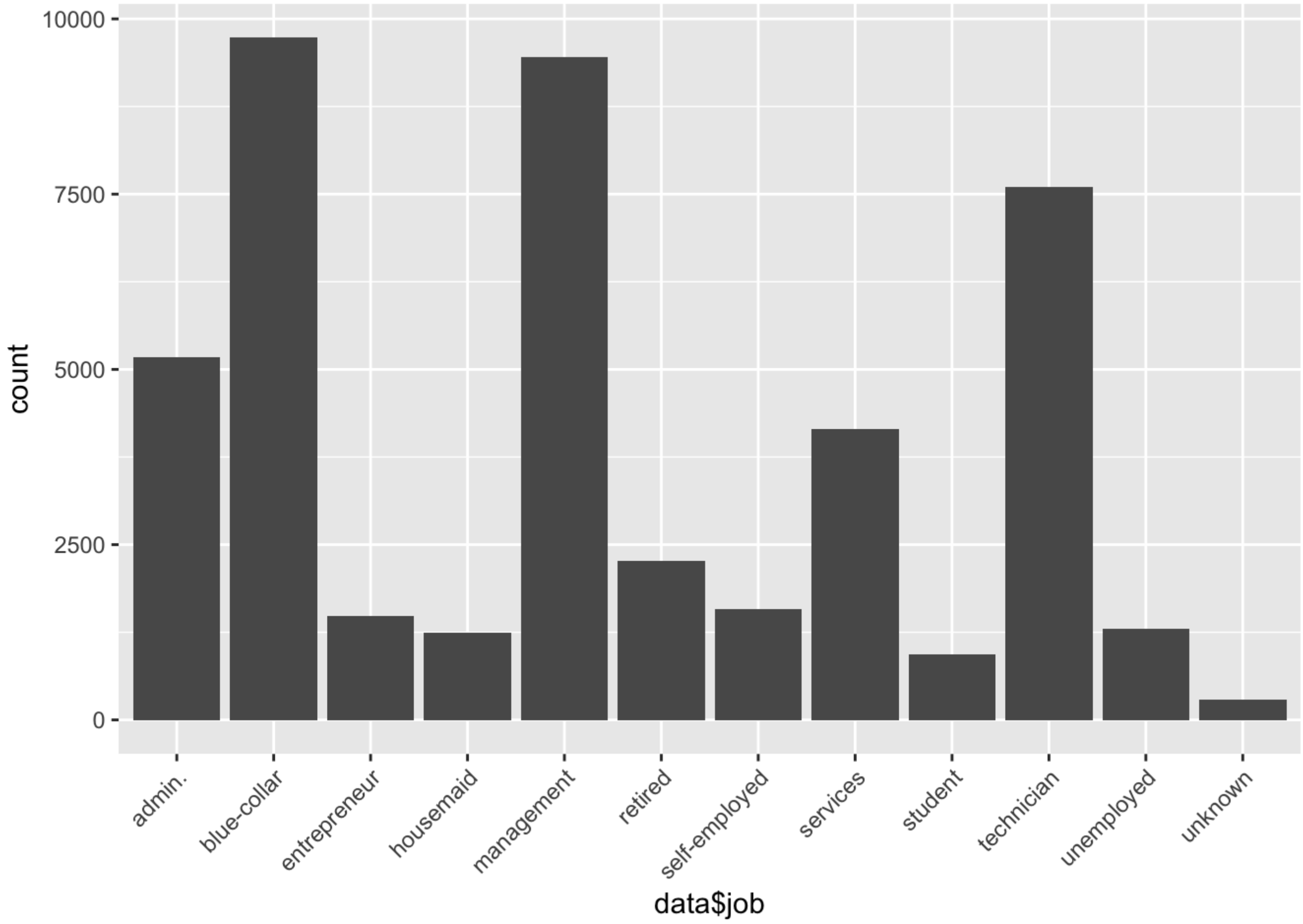
Subscription percentage of divorced and married customers are a little below the general average while single and unknown marital status customers subscription percentage is almost 3% higher then average.

Visualizing the Distribution of the Variables

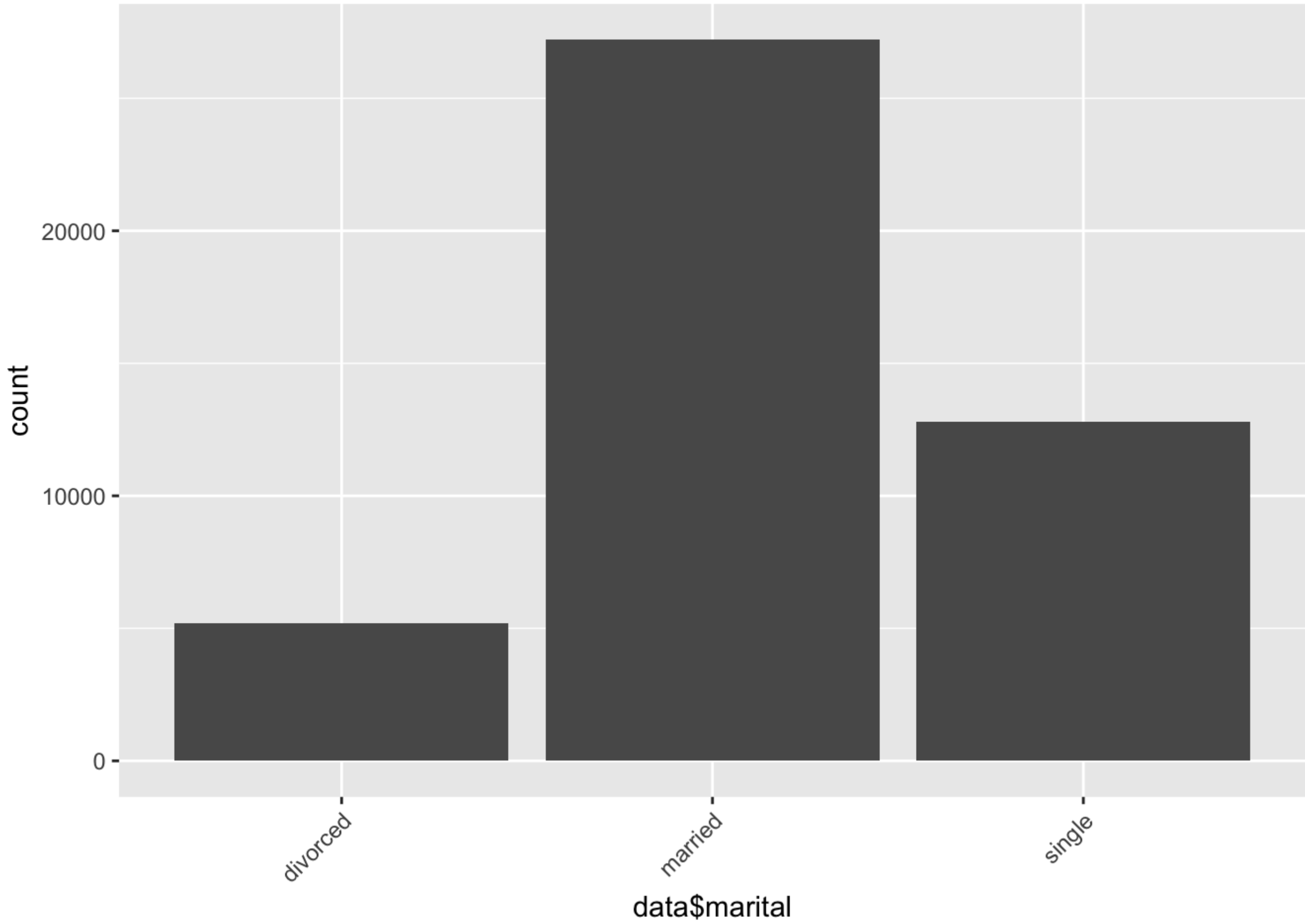
```
data %>%
    ggplot(aes(education)) +
    geom_bar(aes(fill=y)) +
    theme( axis.text.x = element_text(angle = 65,vjust = 0.5, hjust = 0.5, size = 12
))
```



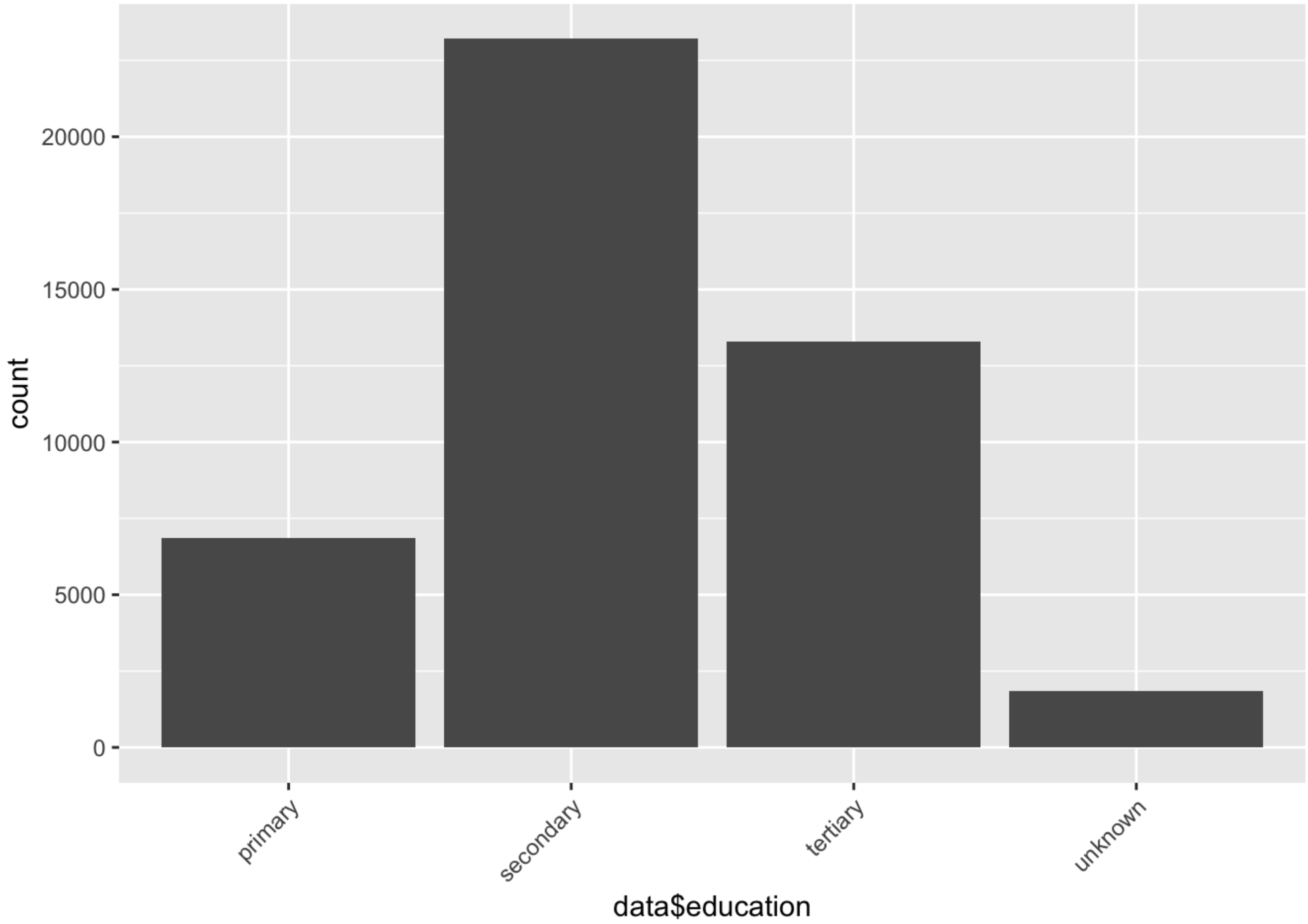

```
data %>%
    ggplot(aes(data$job)) +
    geom_bar() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



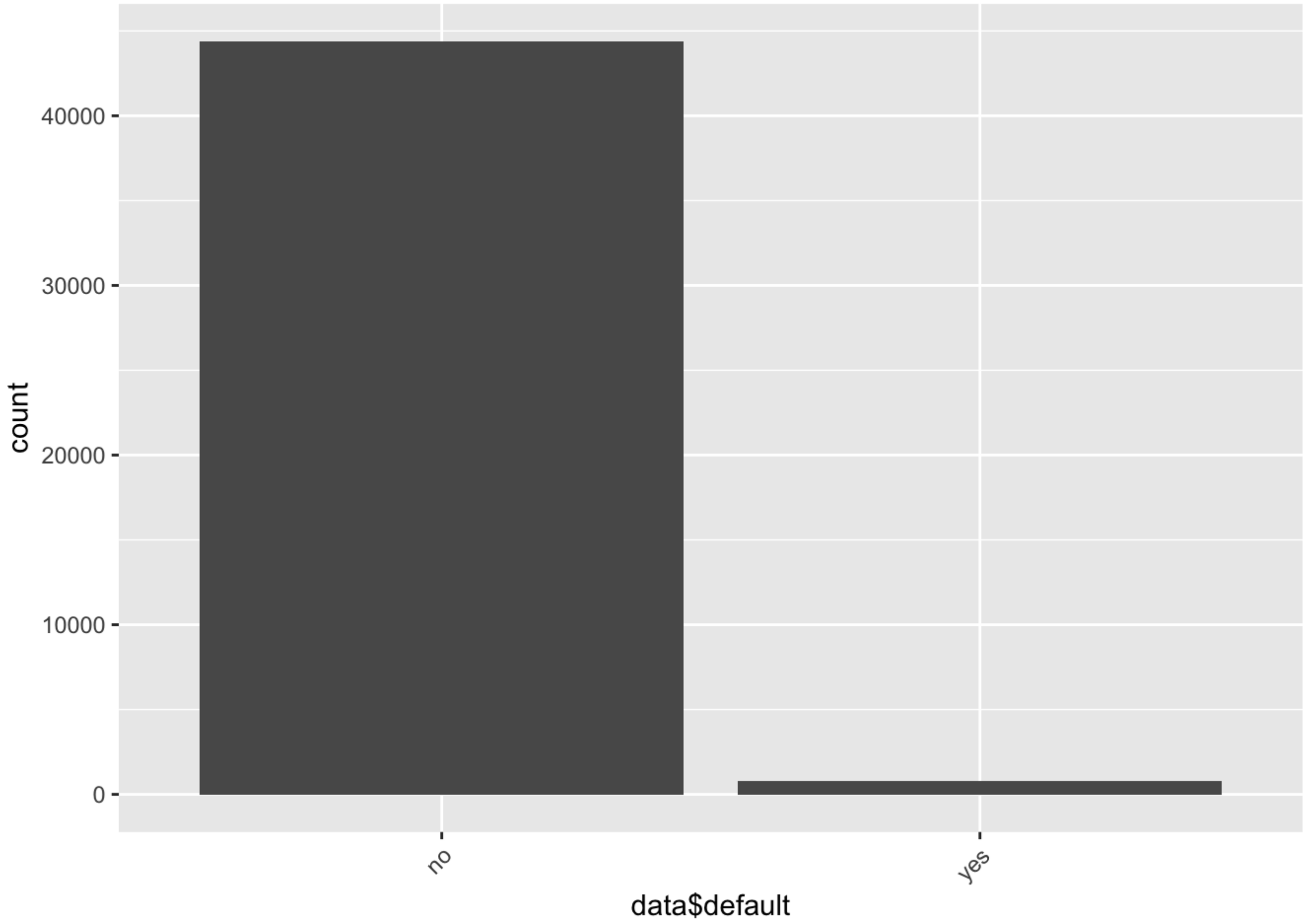

```
data %>%
    ggplot(aes(data$marital)) +
    geom_bar() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



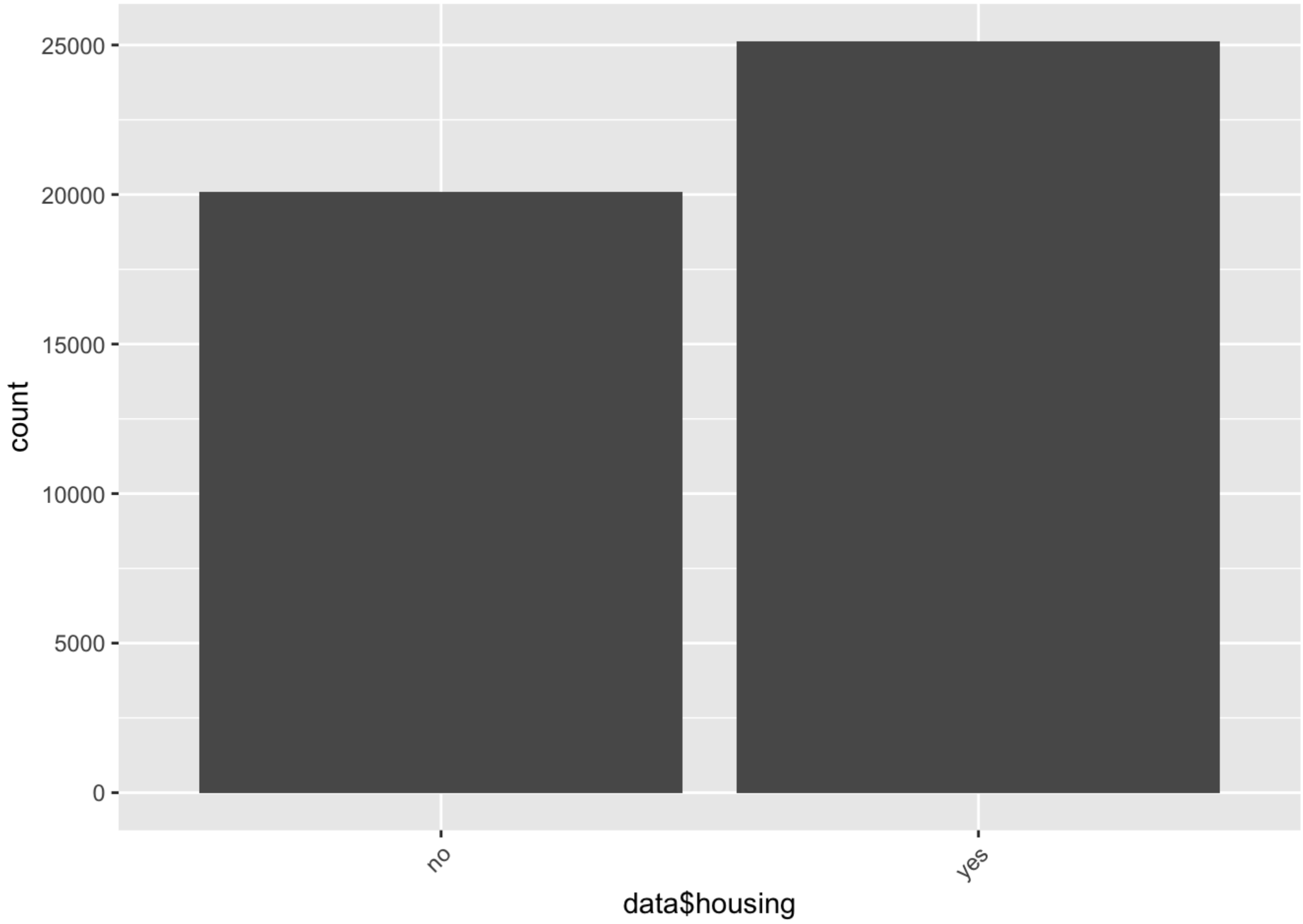

```
data %>%
    ggplot(aes(data$education)) +
    geom_bar() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



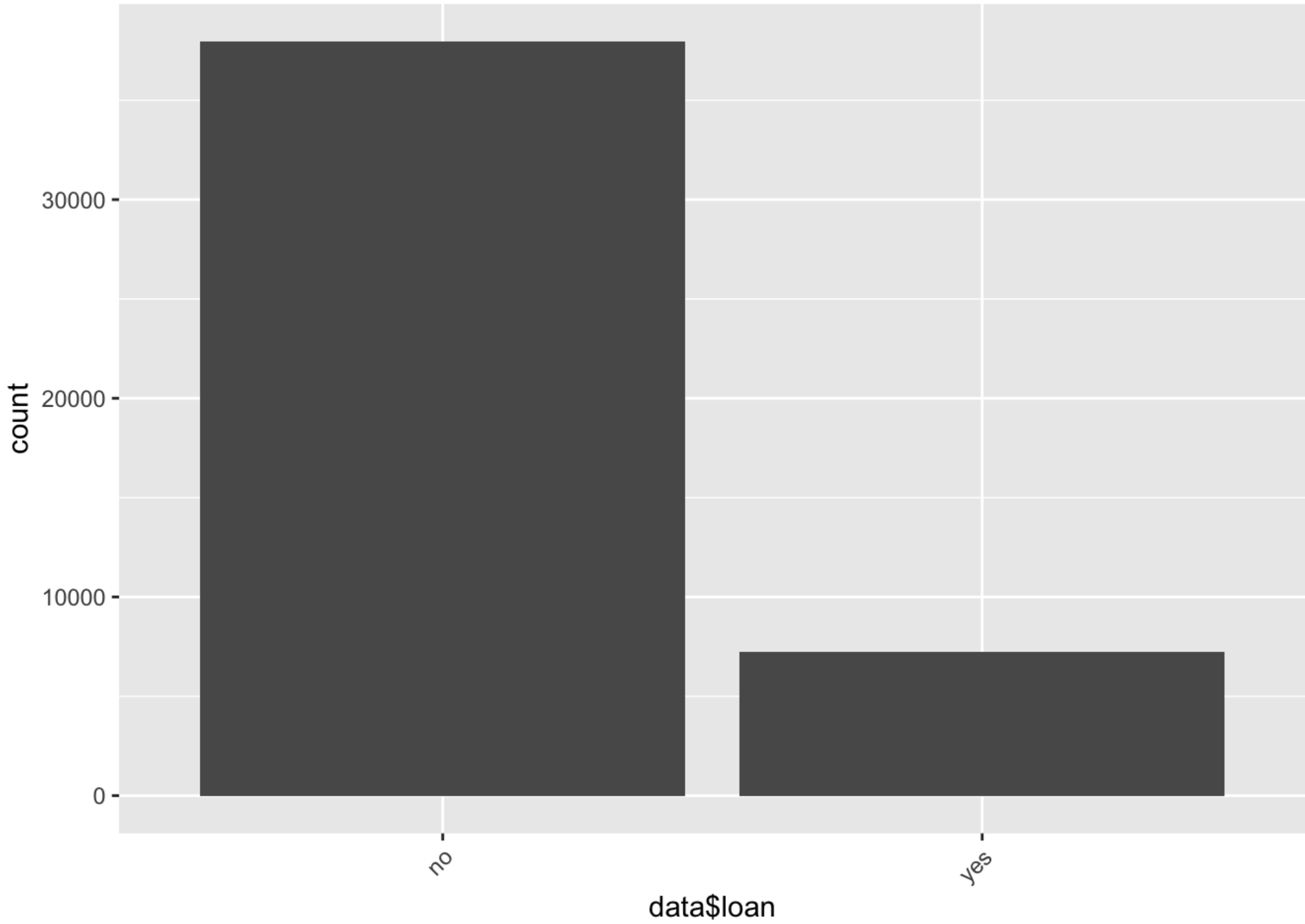

```
data %>%
    ggplot(aes(data$default)) +
    geom_bar() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



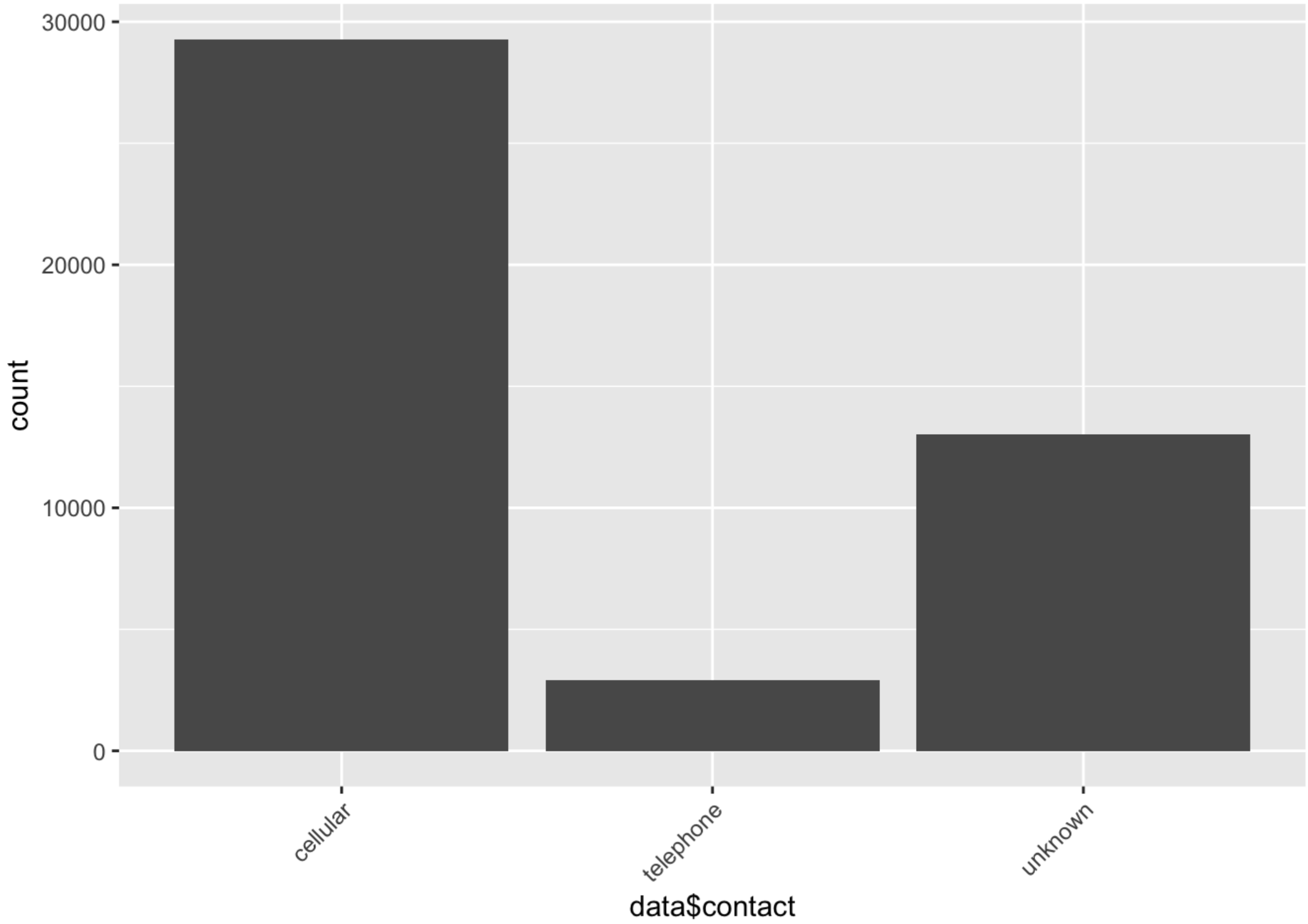

```
data %>%
    ggplot(aes(data$housing)) +
    geom_bar() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



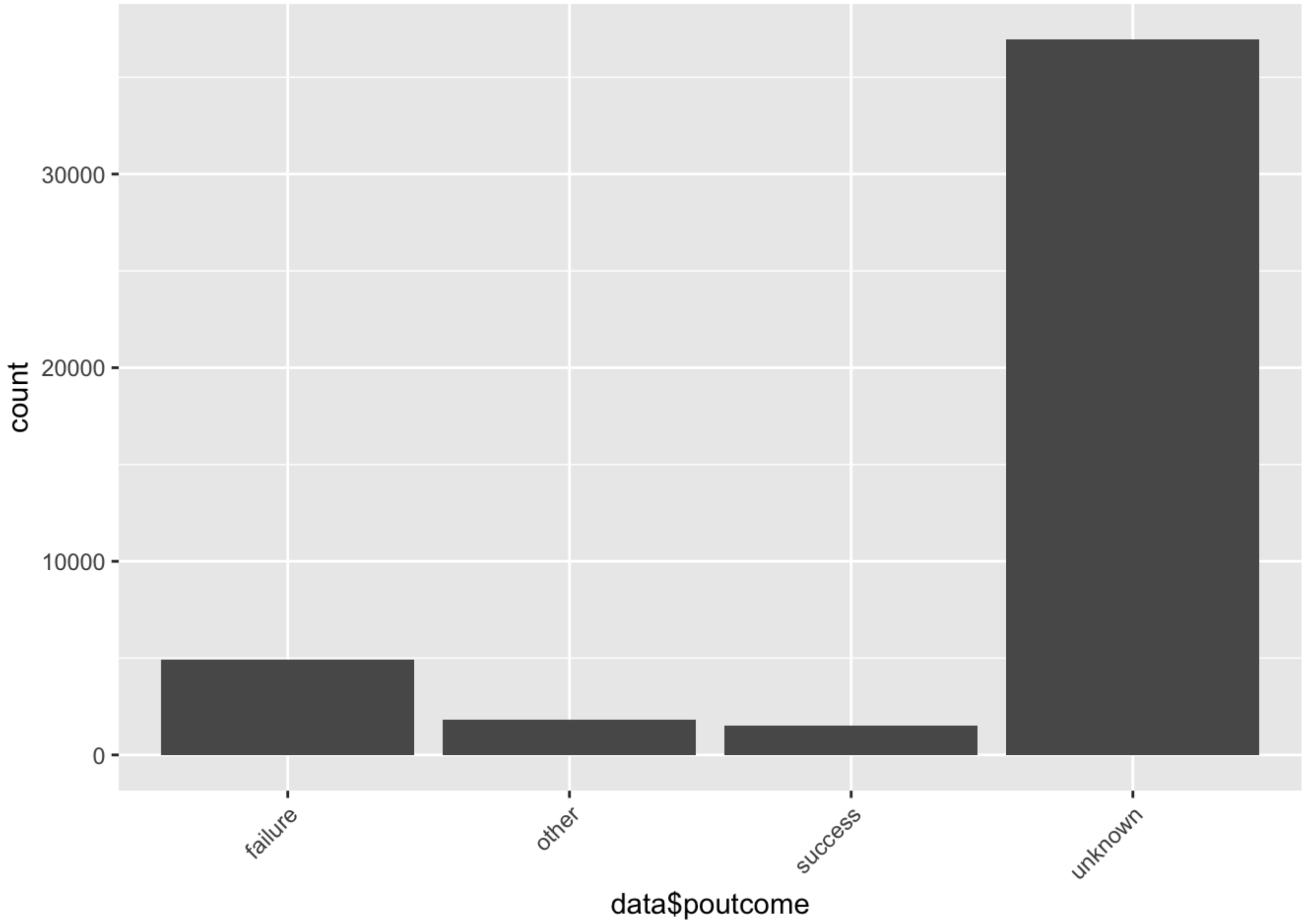

```
data %>%
    ggplot(aes(data$loan)) +
    geom_bar() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



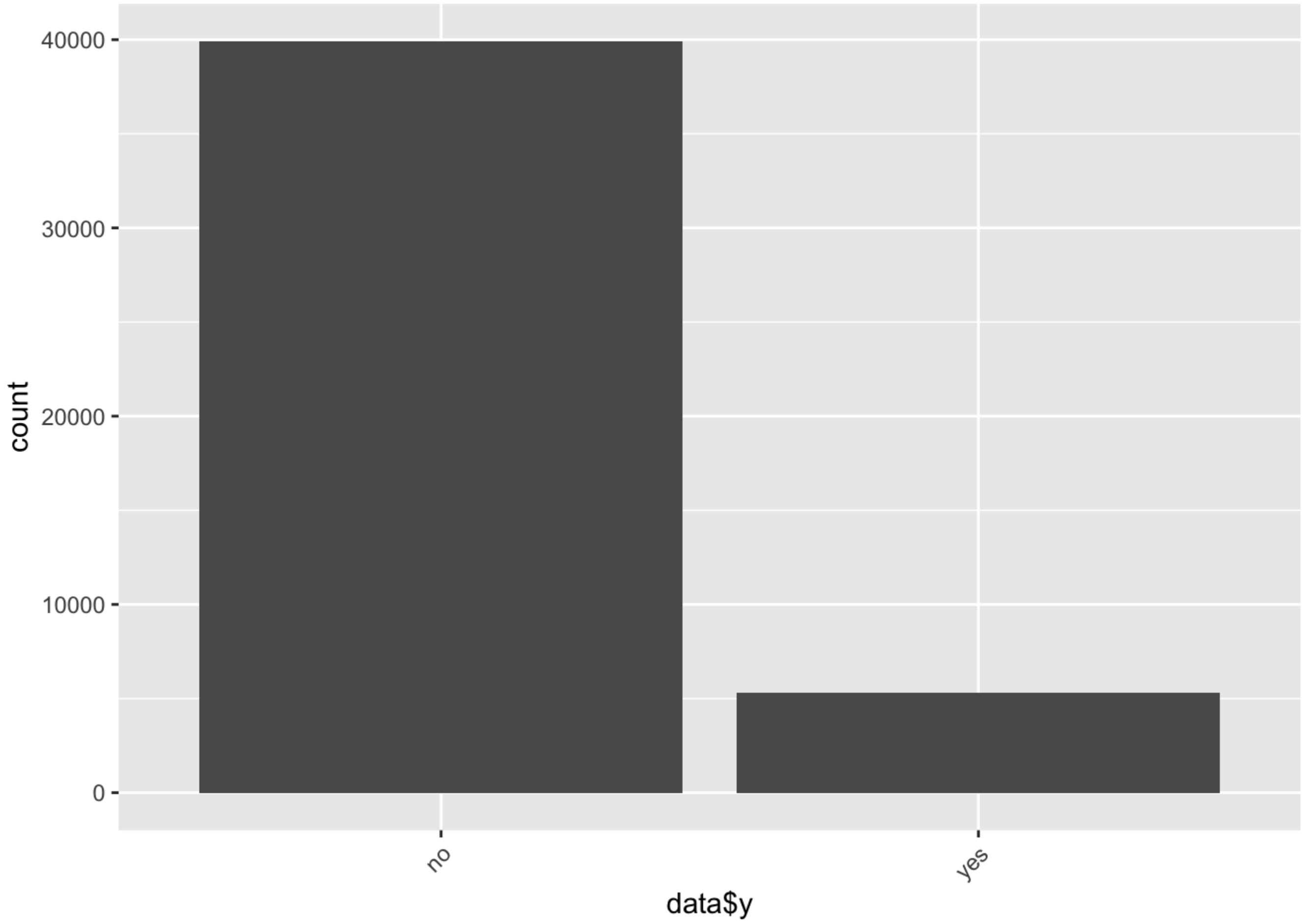

```
data %>%
    ggplot(aes(data$contact)) +
    geom_bar() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



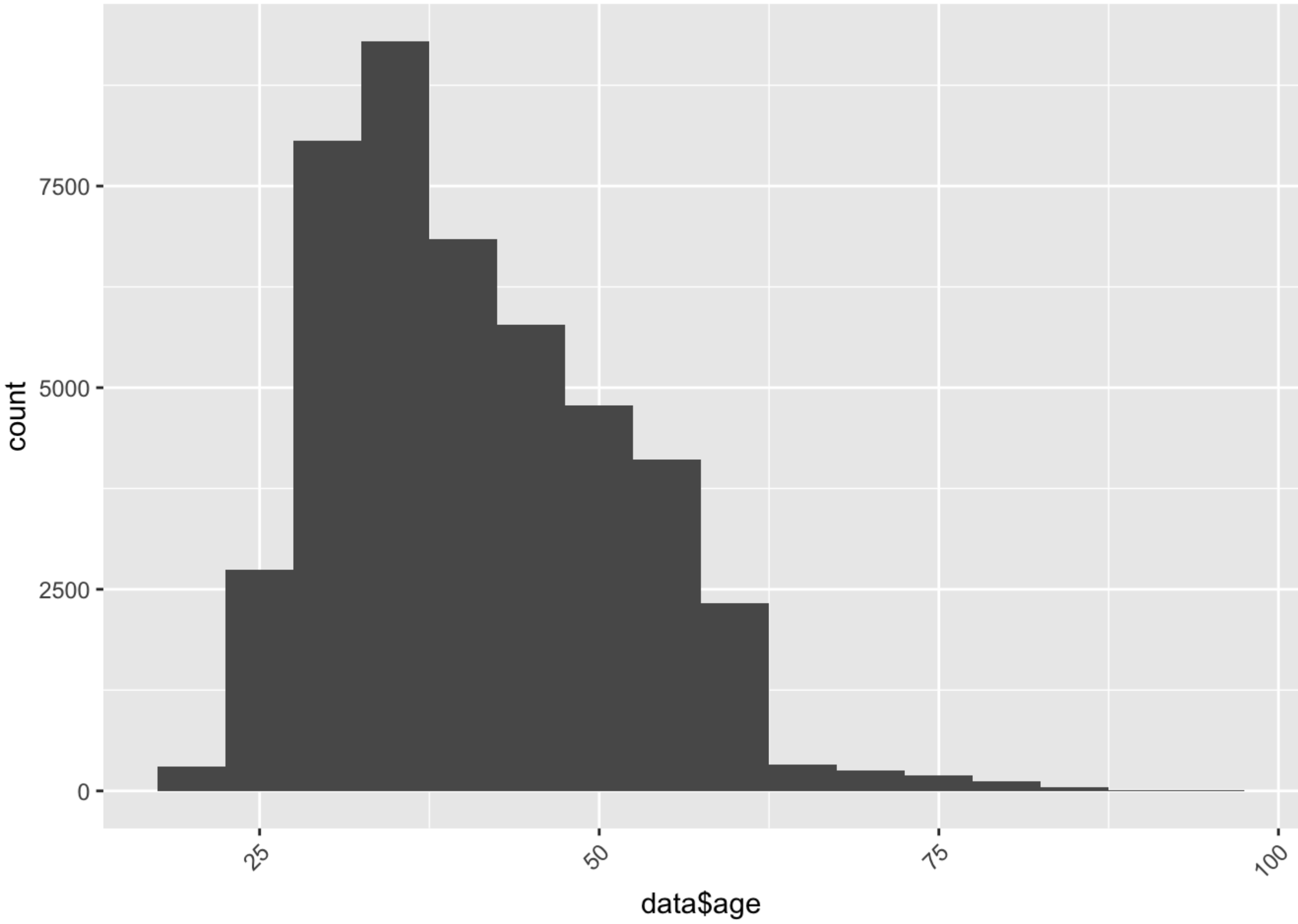

```
data %>%
    ggplot(aes(data$poutcome)) +
    geom_bar() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



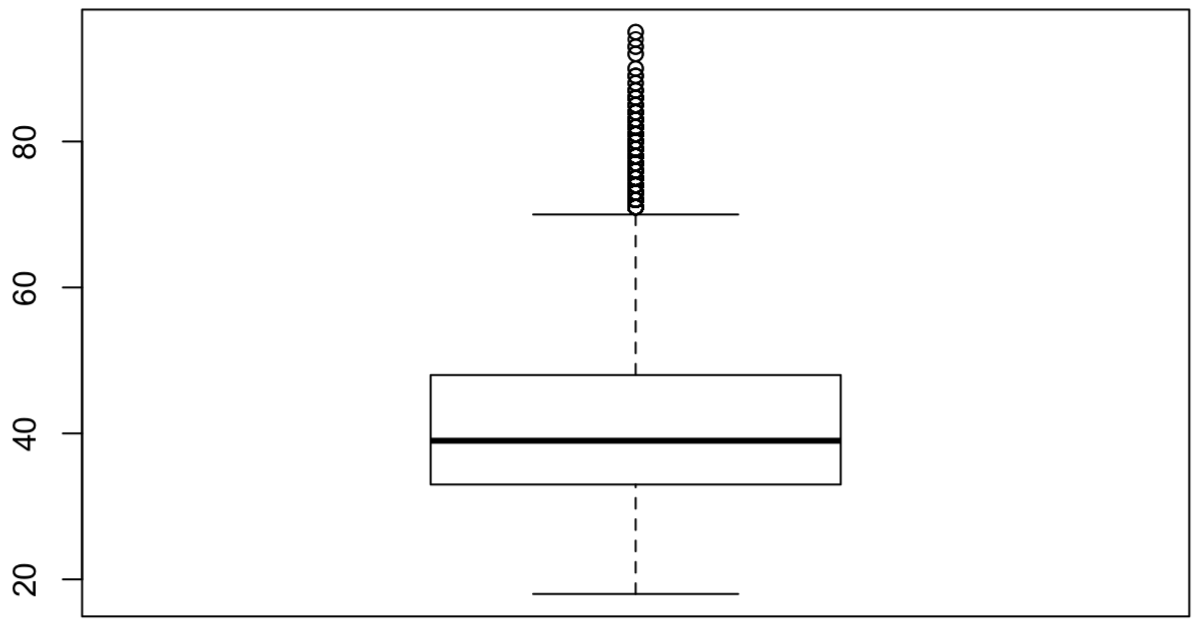

```
data %>%
    ggplot(aes(data$y)) +
    geom_bar() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



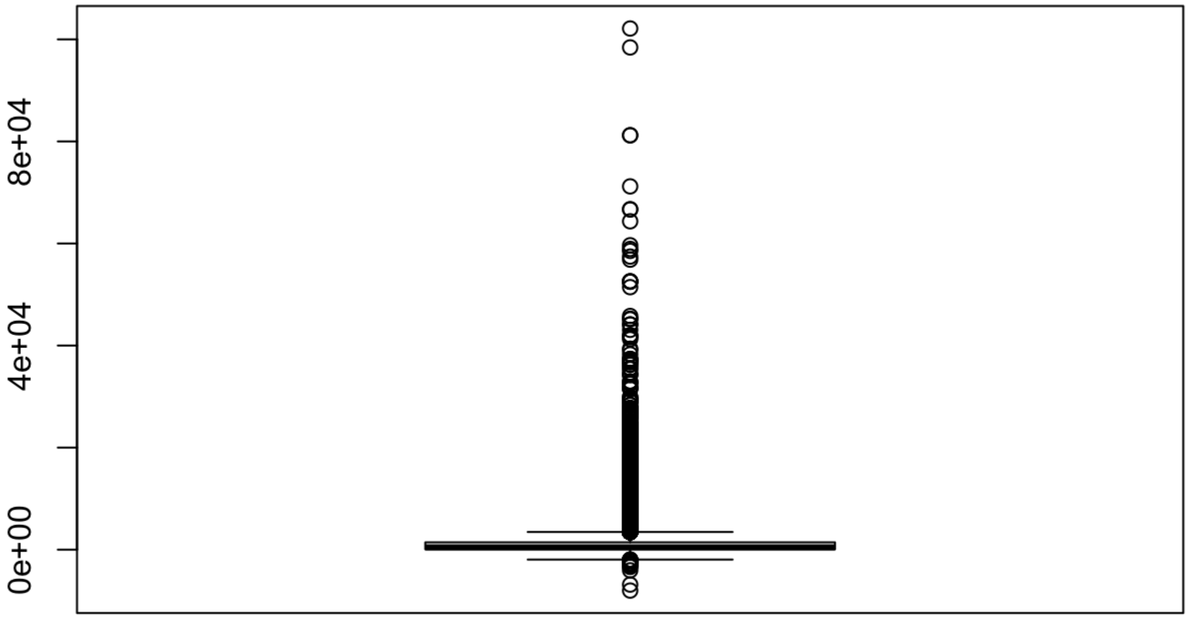

```
data %>%
    ggplot(aes(data$age)) +
    geom_histogram(binwidth = 5) +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



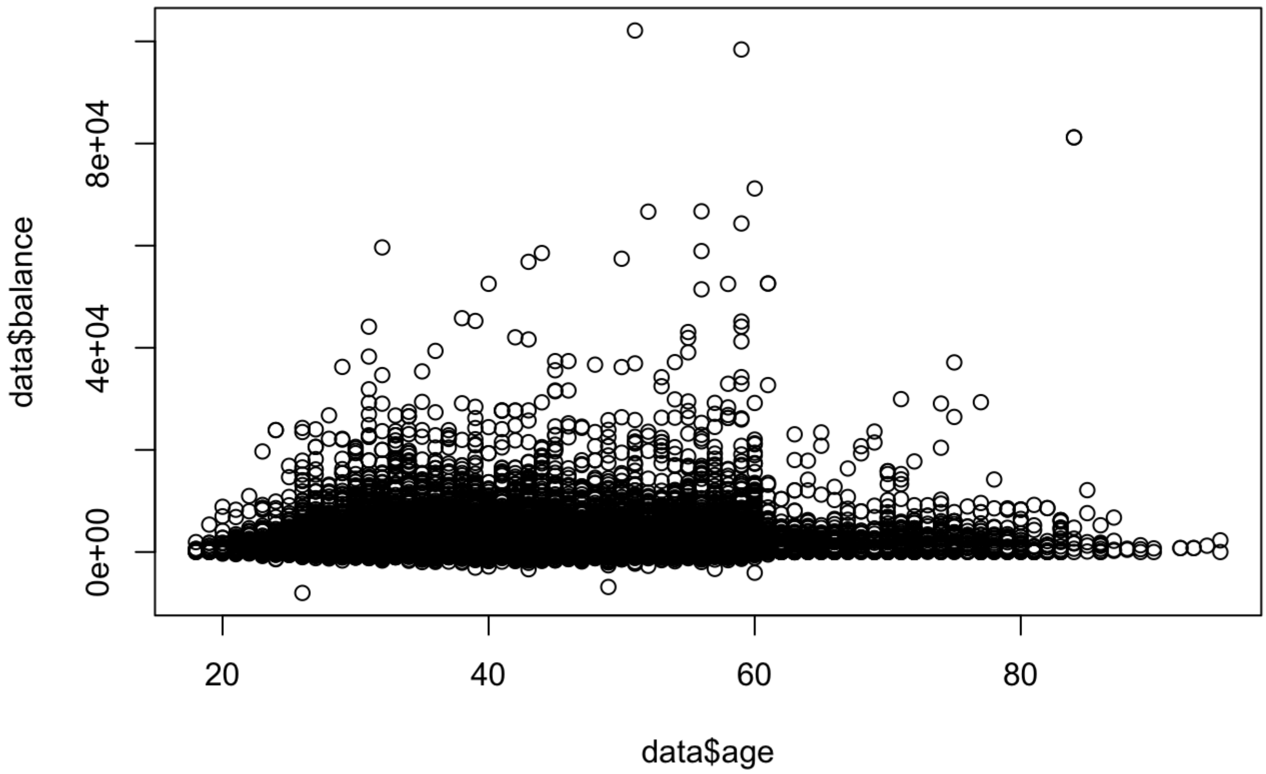

```
boxplot(data$age)
```



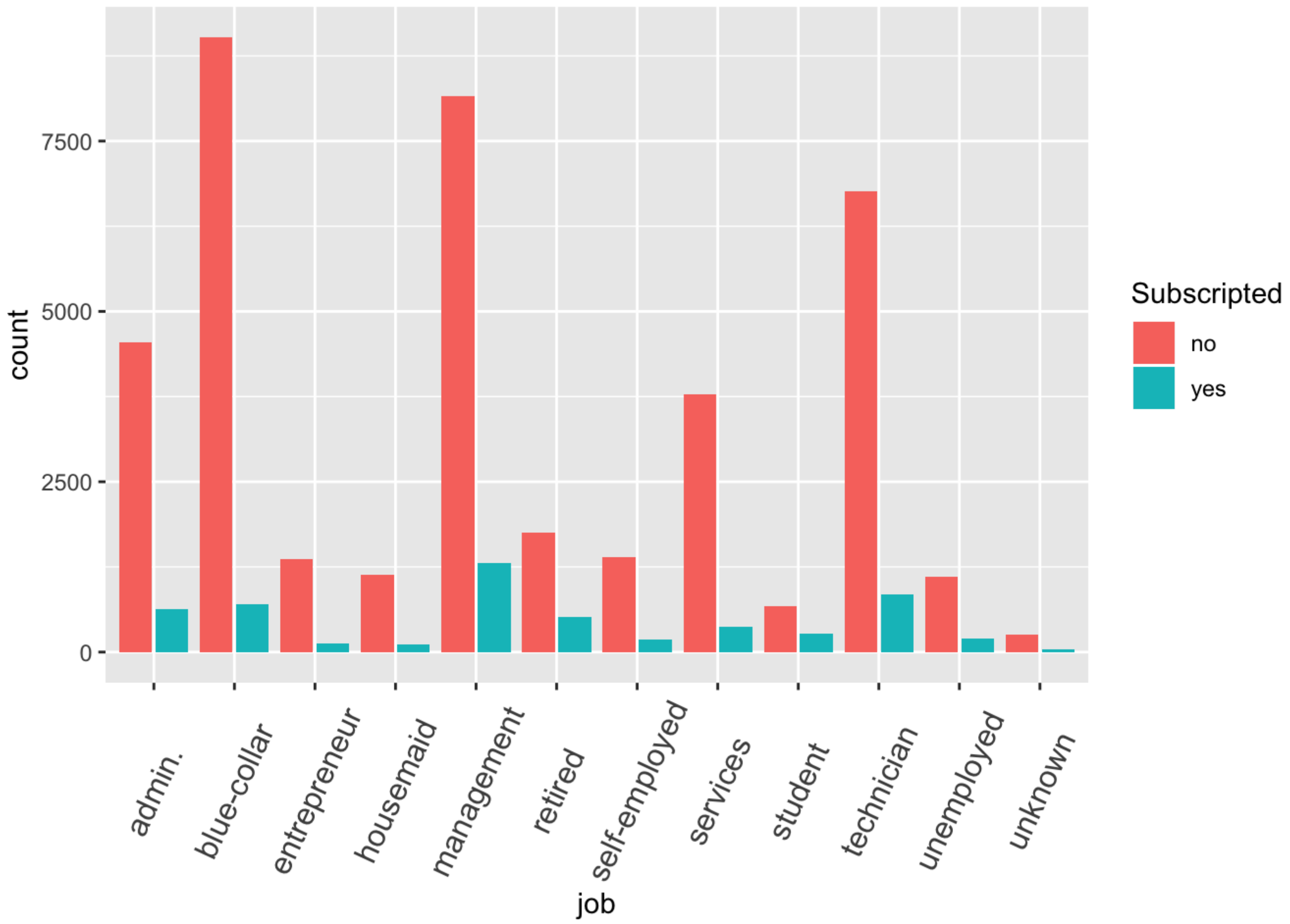
```
boxplot(data$balance)
```



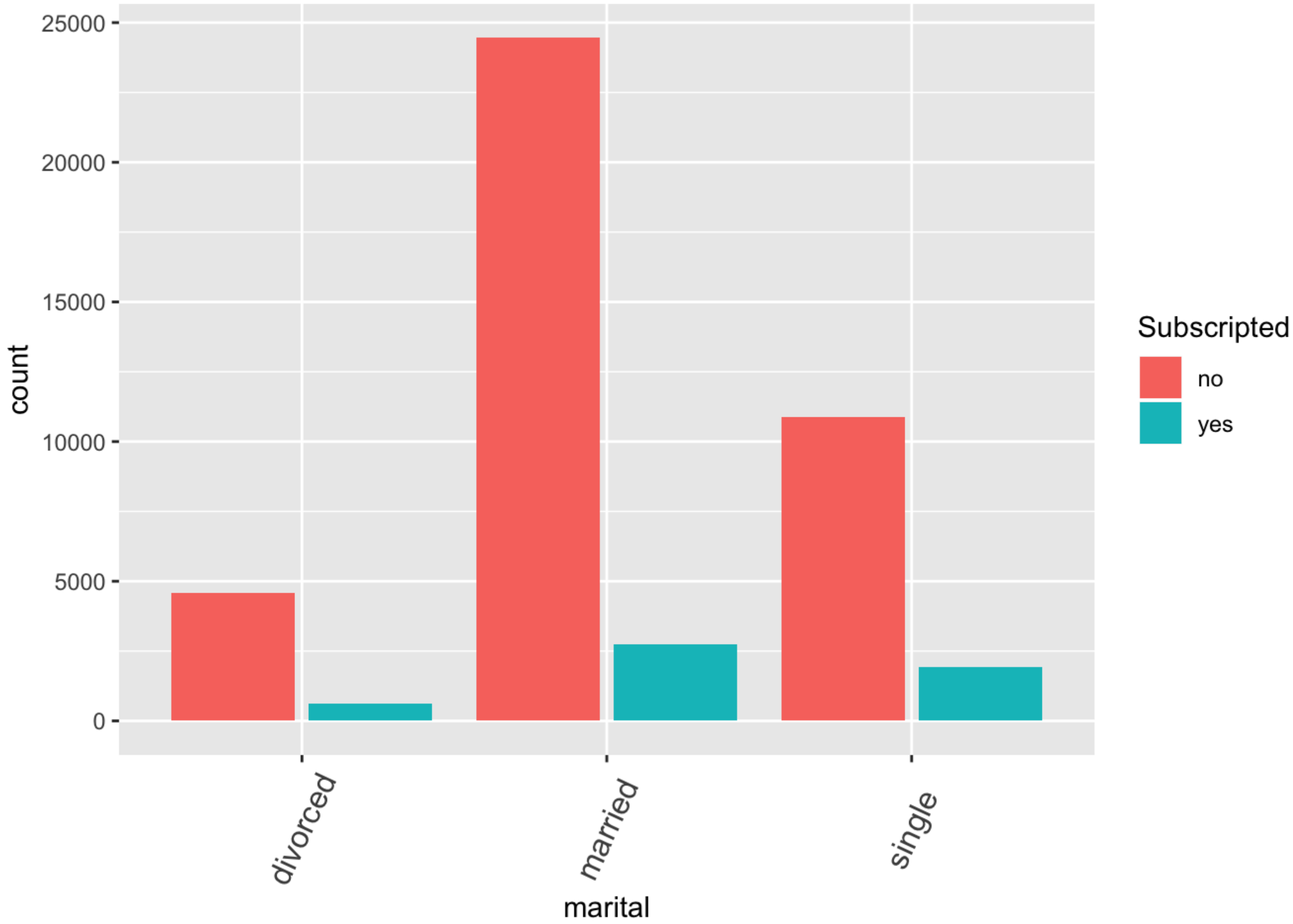

```
plot(data$age, data$balance)
```



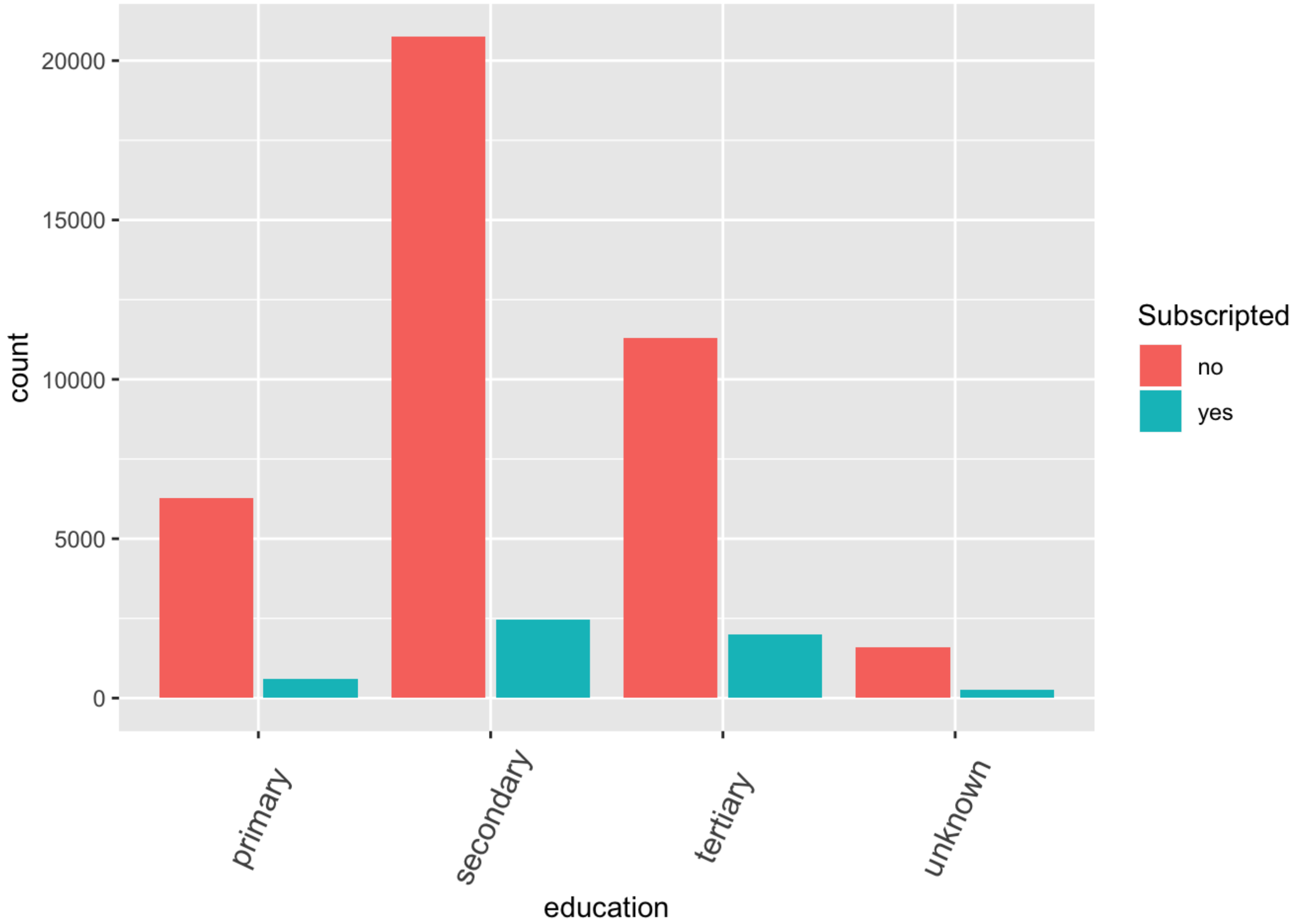

```
data %>%
    ggplot(aes(x=job, fill=y))+
    geom_bar(position="dodge2")+
    guides(fill=guide_legend(title="Subscripted")) +
    theme( axis.text.x = element_text(angle = 65,vjust = 0.5, hjust = 0.5, size = 12
))
```



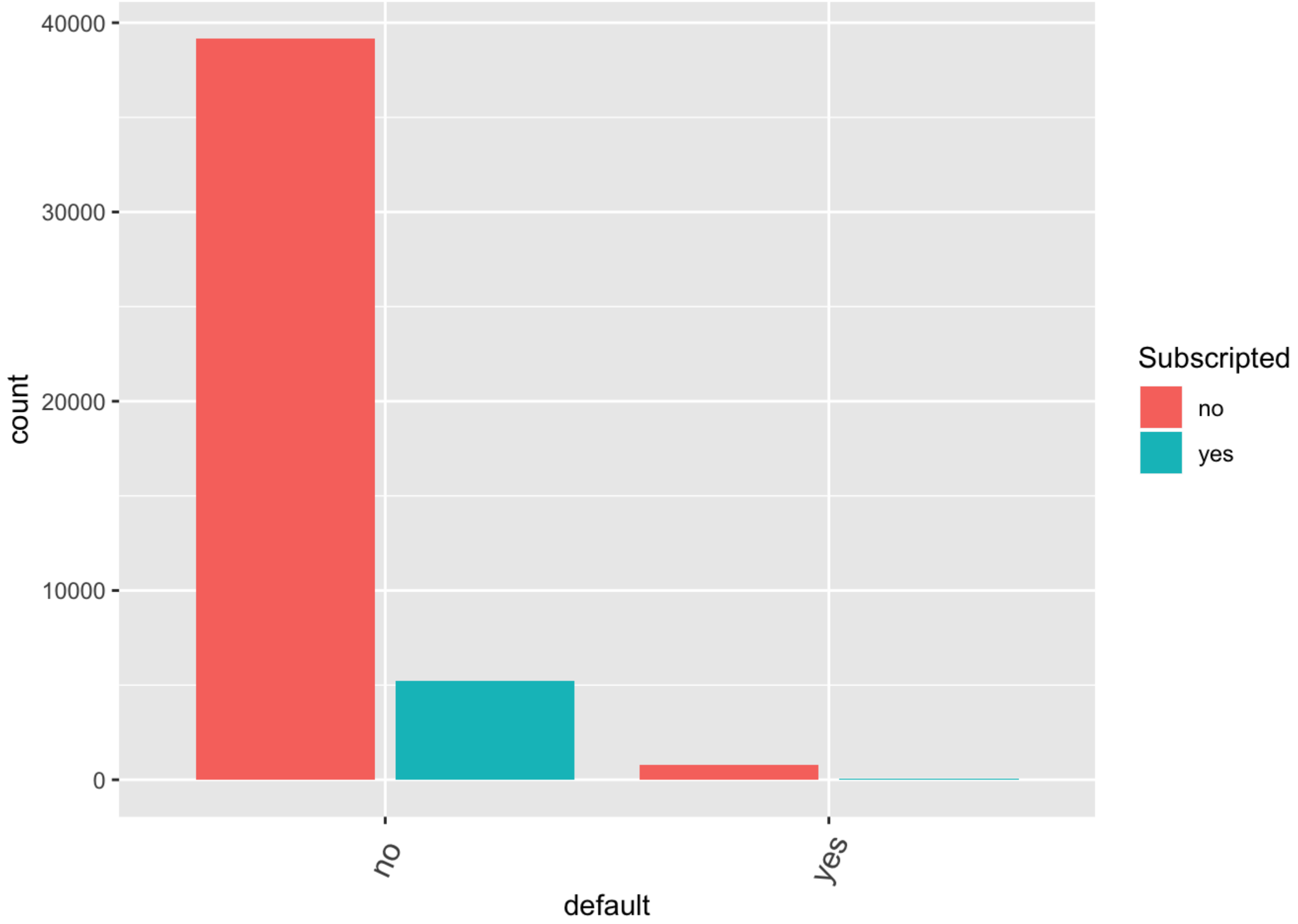

```
data %>%
    ggplot(aes(x=marital, fill=y))+
    geom_bar(position="dodge2")+
    guides(fill=guide_legend(title="Subscripted")) +
    theme( axis.text.x = element_text(angle = 65,vjust = 0.5, hjust = 0.5, size = 12
))
```




```
data %>%
    ggplot(aes(x=education, fill=y))+
    geom_bar(position="dodge2")+
    guides(fill=guide_legend(title="Subscripted")) +
    theme( axis.text.x = element_text(angle = 65,vjust = 0.5, hjust = 0.5, size = 12
))
```




```
data %>%
    ggplot(aes(x=default, fill=y))+
    geom_bar(position="dodge2")+
    guides(fill=guide_legend(title="Subscripted")) +
    theme( axis.text.x = element_text(angle = 65,vjust = 0.5, hjust = 0.5, size = 12
))
```




```
data %>%
    group_by(job) %>%
    summarise(yes=sum(y=="yes"), no= sum(y=="no"),
              yes_pct=round(yes*100/(yes+no),2),
              no_pct=round(no*100/(yes+no),2))%>%
    arrange(job) %>%
    select(-yes, -no)
```

```
## # A tibble: 12 x 3
##    job           yes_pct no_pct
##    <fct>           <dbl>  <dbl>
##  1 admin.          12.2    87.8
##  2 blue-collar      7.27   92.7
##  3 entrepreneur     8.27   91.7
##  4 housemaid        8.79   91.2
##  5 management      13.8    86.2
##  6 retired         22.8    77.2
##  7 self-employed   11.8    88.2
##  8 services         8.88   91.1
##  9 student         28.7    71.3
## 10 technician      11.1    88.9
## 11 unemployed      15.5    84.5
## 12 unknown         11.8    88.2
```

```
groups <- c(quo(job), quo(marital), quo(education), quo(default),
            quo(housing), quo(loan), quo(contact), quo(poutcome))

for (i in seq_along(groups)) {
  data %>%
    group_by(!!groups[[i]]) %>%
    summarise(yes=sum(y=="yes"), no= sum(y=="no"),
              yes_pct=round(yes*100/(yes+no),2),
              no_pct=round(no*100/(yes+no),2))%>%
    arrange(!!groups[[i]]) %>%
    select(-yes, -no) %>%
    print()
    cat('\n')
}
```

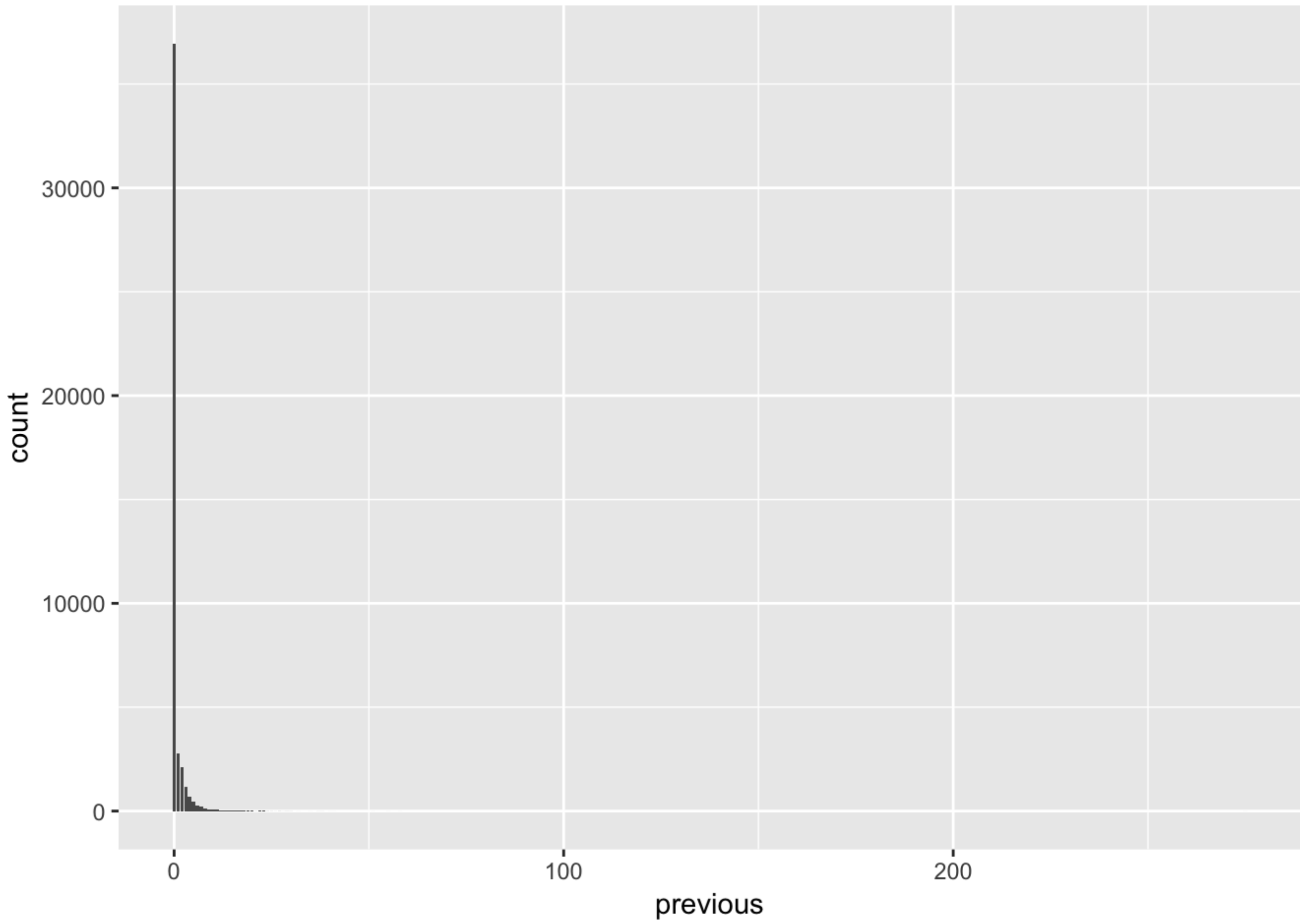
```
## # A tibble: 12 x 3
##    job           yes_pct no_pct
##    <fct>           <dbl>  <dbl>
##  1 admin.          12.2    87.8
##  2 blue-collar      7.27   92.7
##  3 entrepreneur     8.27   91.7
##  4 housemaid        8.79   91.2
##  5 management      13.8    86.2
##  6 retired         22.8    77.2
##  7 self-employed   11.8    88.2
##  8 services         8.88   91.1
##  9 student         28.7    71.3
## 10 technician      11.1    88.9
## 11 unemployed      15.5    84.5
## 12 unknown         11.8    88.2
##
## # A tibble: 3 x 3
##   marital  yes_pct no_pct
##   <fct>      <dbl>  <dbl>
## 1 divorced    12.0   88.0
## 2 married     10.1   89.9
## 3 single      15.0   85.0
##
## # A tibble: 4 x 3
##   education yes_pct no_pct
##   <fct>       <dbl>  <dbl>
## 1 primary      8.63   91.4
## 2 secondary   10.6    89.4
## 3 tertiary    15.0    85.0
## 4 unknown     13.6    86.4
##
## # A tibble: 2 x 3
##   default yes_pct no_pct
##   <fct>     <dbl>  <dbl>
## 1 no        11.8    88.2
## 2 yes        6.38   93.6
##
## # A tibble: 2 x 3
##   housing yes_pct no_pct
##   <fct>     <dbl>  <dbl>
## 1 no         16.7   83.3
## 2 yes         7.7   92.3
##
## # A tibble: 2 x 3
##   loan  yes_pct no_pct
##   <fct>   <dbl>  <dbl>
## 1 no      12.7    87.3
## 2 yes      6.68   93.3
##
## # A tibble: 3 x 3
##   contact   yes_pct no_pct
##   <fct>       <dbl>  <dbl>
## 1 cellular    14.9    85.1
## 2 telephone   13.4    86.6
## 3 unknown      4.07   95.9
##
## # A tibble: 4 x 3
```

```
##   poutcome yes_pct no_pct
##   <fct>      <dbl>  <dbl>
## 1 failure    12.6    87.4
## 2 other      16.7    83.3
## 3 success    64.7    35.3
## 4 unknown     9.16   90.8
```

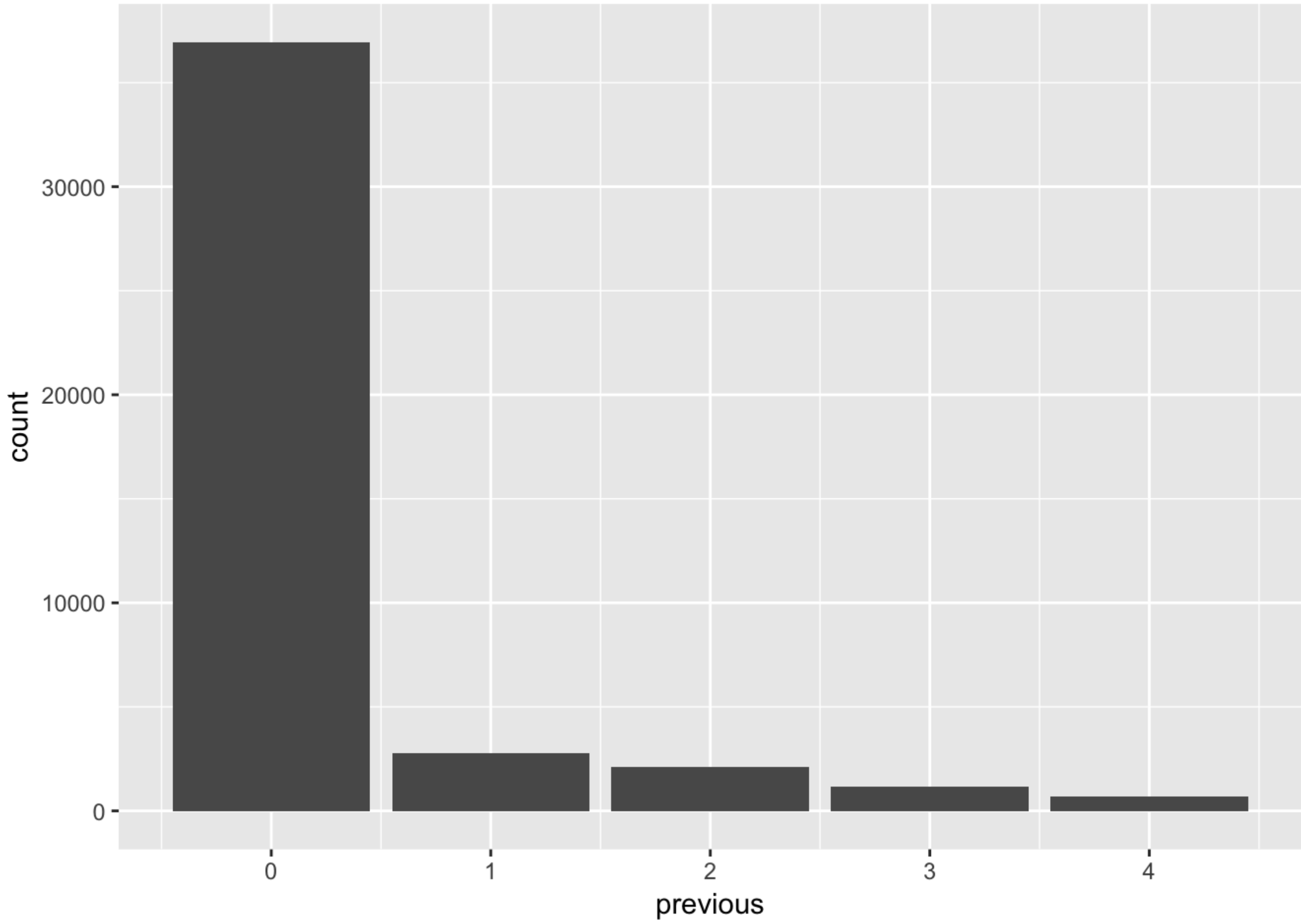
We decided to remove the “day” and “pdays” columns as they are irrelevant for the analysis.

```
data = data %>%
    select(-day, -pdays)
```

```
data %>%
    ggplot(aes(previous)) +
    geom_bar()
```




```
data %>%
    filter(previous < 5) %>%
    ggplot(aes(previous)) +
    geom_bar()
```



Since 82% of all observations in “previous” column are zero, we decided to convert it to binary which translates to 0: not contacted before and 1: contacted before.

```
data = data %>%
    mutate(previous = ifelse(previous == 0, 0, 1))
data$previous = as.integer(data$previous)
```

```
num_vars <- c('age', 'balance', 'duration', 'campaign')
```

Outliers in the dataset were detected based on IQR rule.

```
Outliers <- c()

for(i in num_vars){

  max <- quantile(data[,i],0.75, na.rm=TRUE) + (IQR(data[,i], na.rm=TRUE) * 3 )
  min <- quantile(data[,i],0.25, na.rm=TRUE) - (IQR(data[,i], na.rm=TRUE) * 3 )

  idx <- which(data[,i] < min | data[,i] > max)

  print(paste(i, length(idx), sep=' : '))

  Outliers <- c(Outliers, idx)
}
```

```
## [1] "age : 3"
## [1] "balance : 2443"
## [1] "duration : 1155"
## [1] "campaign : 1462"
```

```
Outliers <- sort(Outliers)

data <- data[-Outliers,]
```

Target variable “y” and other binary variables were transformed into numerical type.

```
data$y <- as.integer(as.character(factor(data$y, levels = c("no", "yes"), labels = c(
"0", "1"))))
```

```
data$default <- as.integer(as.character(factor(data$default, levels = c("no", "yes"),
labels = c("0", "1"))))
data$housing <- as.integer(as.character(factor(data$housing, levels = c("no", "yes"),
labels = c("0", "1"))))
data$loan <- as.integer(as.character(factor(data$loan, levels = c("no", "yes"), label
s = c("0", "1"))))
```

```
multi = c('job', 'marital', 'education', 'contact', 'poutcome', 'month')
```

Categorical variables that have more than two distinct values were dummified in order to do a more precise analysis.

```
data = dummy.data.frame(data, multi, drop = FALSE)
```

Numerical variables were scaled.

```
data$age = scale(data$age)
data$balance = scale(data$balance)
data$duration = scale(data$duration)
data$campaign = scale(data$campaign)
```

```
str(data)
```

```
## 'data.frame':    40314 obs. of  47 variables:
##  $ age               : num [1:40314, 1] 1.633 0.306 -0.737 0.59 -0.737 ...
##   ..- attr(*, "scaled:center")= num 40.8
##   ..- attr(*, "scaled:scale")= num 10.5
##  $ jobadmin.         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ jobblue-collar    : int  0 0 0 1 0 0 0 0 0 0 ...
##  $ jobentrepreneur   : int  0 0 1 0 0 0 0 1 0 0 ...
##  $ jobhousemaid      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ jobmanagement     : int  1 0 0 0 0 1 1 0 0 0 ...
##  $ jobretired        : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ jobself-employed  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ jobservices       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ jobstudent        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ jobtechnician     : int  0 1 0 0 0 0 0 0 0 1 ...
##  $ jobunemployed     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ jobunknown        : int  0 0 0 0 1 0 0 0 0 0 ...
##  $ maritaldivorced   : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ maritalmarried    : int  1 0 1 1 0 1 0 0 1 0 ...
##  $ maritalsingle     : int  0 1 0 0 1 0 1 0 0 1 ...
##  $ educationprimary  : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ educationsecondary: int  0 1 1 0 0 0 0 0 0 1 ...
##  $ educationtertiary : int  1 0 0 0 0 1 1 1 0 0 ...
##  $ educationunknown  : int  0 0 0 1 1 0 0 0 0 0 ...
##  $ default           : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ balance           : num [1:40314, 1] 1.111 -0.687 -0.71 0.569 -0.71 ...
##   ..- attr(*, "scaled:center")= num 837
##   ..- attr(*, "scaled:scale")= num 1176
##  $ housing           : int  1 1 1 1 0 1 1 1 1 1 ...
##  $ loan              : int  0 0 1 0 0 0 1 0 0 0 ...
##  $ contactcellular   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ contacttelephone  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ contactunknown    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ monthapr          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthaug          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthdec          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthfeb          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthjan          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthjul          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthjun          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthmar          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthmay          : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ monthnov          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthoct          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ monthsep          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ duration          : num [1:40314, 1] 0.153 -0.447 -0.855 -0.768 -0.19 ...
##   ..- attr(*, "scaled:center")= num 233
##   ..- attr(*, "scaled:scale")= num 183
##  $ campaign          : num [1:40314, 1] -0.798 -0.798 -0.798 -0.798 -0.798 ...
##   ..- attr(*, "scaled:center")= num 2.34
##   ..- attr(*, "scaled:scale")= num 1.67
##  $ previous          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ poutcomefailure   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ poutcomeother     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ poutcomesuccess   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ poutcomeunknown   : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ y                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  - attr(*, "dummies")=List of 6
```

```
##   ..$ job      : int  2 3 4 5 6 7 8 9 10 11 ...
##   ..$ marital  : int  14 15 16
##   ..$ education: int  17 18 19 20
##   ..$ contact  : int  25 26 27
##   ..$ month    : int  28 29 30 31 32 33 34 35 36 37 ...
##   ..$ poutcome : int  43 44 45 46
```

Marketing Analytics Applications

Train-Test Split

```
smp_size <- floor(0.75 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
test <- data[-train_ind, ]
```

Logistic Regression

-Fitting the model-

```
model <- glm(y ~ ., data = train, family = binomial)
```

```
summary(model)
```

```
##
## Call:
## glm(formula = y ~ ., family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1318  -0.3438  -0.2153  -0.1251   3.4042
##
## Coefficients: (6 not defined because of singularities)
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -2.531978   0.324333  -7.807 5.87e-15 ***
## age                -0.002235   0.030207  -0.074 0.941027
## jobadmin.           0.315088   0.293597   1.073 0.283181
## `jobblue-collar`    0.018370   0.292982   0.063 0.950004
## jobentrepreneur    -0.149204   0.323086  -0.462 0.644217
## jobhousemaid       -0.194128   0.325367  -0.597 0.550745
## jobmanagement       0.216024   0.291421   0.741 0.458525
## jobretired          0.737298   0.297078   2.482 0.013071 *
## `jobself-employed` -0.079256   0.313054  -0.253 0.800137
## jobservices         0.080595   0.299137   0.269 0.787604
## jobstudent          0.897028   0.307659   2.916 0.003549 **
## jobtechnician       0.184034   0.291439   0.631 0.527736
## jobunemployed       0.108351   0.312188   0.347 0.728538
## jobunknown                NA         NA      NA       NA
## maritaldivorced    -0.126452   0.087937  -1.438 0.150440
## maritalmarried     -0.207266   0.059195  -3.501 0.000463 ***
## maritalsingle             NA         NA      NA       NA
## educationprimary   -0.396738   0.132606  -2.992 0.002773 **
## educationsecondary -0.121475   0.115558  -1.051 0.293167
## educationtertiary   0.057111   0.121433   0.470 0.638133
## educationunknown          NA         NA      NA       NA
## default             0.098597   0.202840   0.486 0.626908
## balance             0.118362   0.021997   5.381 7.42e-08 ***
## housing            -0.762339   0.056055 -13.600  < 2e-16 ***
## loan               -0.476451   0.076939  -6.193 5.92e-10 ***
## contactcellular     1.673170   0.092764  18.037  < 2e-16 ***
## contacttelephone    1.555208   0.130301  11.936  < 2e-16 ***
## contactunknown            NA         NA      NA       NA
## monthapr           -1.005993   0.149288  -6.739 1.60e-11 ***
## monthaug           -1.610066   0.145194 -11.089  < 2e-16 ***
## monthdec           -0.289381   0.241264  -1.199 0.230357
## monthfeb           -1.145929   0.152065  -7.536 4.85e-14 ***
## monthjan           -2.320292   0.190131 -12.204  < 2e-16 ***
## monthjul           -1.838523   0.148683 -12.365  < 2e-16 ***
## monthjun           -0.450437   0.154265  -2.920 0.003502 **
## monthmar            0.742474   0.186734   3.976 7.01e-05 ***
## monthmay           -1.395089   0.144332  -9.666  < 2e-16 ***
## monthnov           -1.718688   0.152601 -11.263  < 2e-16 ***
## monthoct            0.055588   0.173119   0.321 0.748138
## monthsep                  NA         NA      NA       NA
## duration            1.058531   0.020173  52.473  < 2e-16 ***
## campaign           -0.222311   0.029199  -7.614 2.66e-14 ***
## previous            0.698768   1.421252   0.492 0.622962
## poutcomefailure    -0.537251   1.422072  -0.378 0.705583
## poutcomeother      -0.307016   1.423769  -0.216 0.829271
## poutcomesuccess     1.787126   1.422801   1.256 0.209094
## poutcomeunknown           NA         NA      NA       NA
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 20268  on 30234  degrees of freedom
## Residual deviance: 13058  on 30194  degrees of freedom
## AIC: 13140
##
## Number of Fisher Scoring iterations: 6
```

Since columns maritalsingle, monthsep, jobunknown, educationunknown, contactunknown and poutcomeunknown are highly correlated, the output related to these columns happened to be NA.

```
model <- glm(y ~ .-maritalsingle -monthsep -jobunknown -educationunknown -contactunkn
own -poutcomeunknown,
               data = train, family = binomial)
```

```
summary(model)
```

```
##
## Call:
## glm(formula = y ~ . - maritalsingle - monthsep - jobunknown -
##     educationunknown - contactunknown - poutcomeunknown, family = binomial,
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1318  -0.3438  -0.2153  -0.1251   3.4042
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -2.531978   0.324333  -7.807 5.87e-15 ***
## age                -0.002235   0.030207  -0.074 0.941027
## jobadmin.           0.315088   0.293597   1.073 0.283181
## `jobblue-collar`    0.018370   0.292982   0.063 0.950004
## jobentrepreneur    -0.149204   0.323086  -0.462 0.644217
## jobhousemaid       -0.194128   0.325367  -0.597 0.550745
## jobmanagement       0.216024   0.291421   0.741 0.458525
## jobretired          0.737298   0.297078   2.482 0.013071 *
## `jobself-employed` -0.079256   0.313054  -0.253 0.800137
## jobservices         0.080595   0.299137   0.269 0.787604
## jobstudent          0.897028   0.307659   2.916 0.003549 **
## jobtechnician       0.184034   0.291439   0.631 0.527736
## jobunemployed       0.108351   0.312188   0.347 0.728538
## maritaldivorced    -0.126452   0.087937  -1.438 0.150440
## maritalmarried     -0.207266   0.059195  -3.501 0.000463 ***
## educationprimary   -0.396738   0.132606  -2.992 0.002773 **
## educationsecondary -0.121475   0.115558  -1.051 0.293167
## educationtertiary   0.057111   0.121433   0.470 0.638133
## default             0.098597   0.202840   0.486 0.626908
## balance             0.118362   0.021997   5.381 7.42e-08 ***
## housing            -0.762339   0.056055 -13.600  < 2e-16 ***
## loan               -0.476451   0.076939  -6.193 5.92e-10 ***
## contactcellular     1.673170   0.092764  18.037  < 2e-16 ***
## contacttelephone    1.555208   0.130301  11.936  < 2e-16 ***
## monthapr           -1.005993   0.149288  -6.739 1.60e-11 ***
## monthaug           -1.610066   0.145194 -11.089  < 2e-16 ***
## monthdec           -0.289381   0.241264  -1.199 0.230357
## monthfeb           -1.145929   0.152065  -7.536 4.85e-14 ***
## monthjan           -2.320292   0.190131 -12.204  < 2e-16 ***
## monthjul           -1.838523   0.148683 -12.365  < 2e-16 ***
## monthjun           -0.450437   0.154265  -2.920 0.003502 **
## monthmar            0.742474   0.186734   3.976 7.01e-05 ***
## monthmay           -1.395089   0.144332  -9.666  < 2e-16 ***
## monthnov           -1.718688   0.152601 -11.263  < 2e-16 ***
## monthoct            0.055588   0.173119   0.321 0.748138
## duration            1.058531   0.020173  52.473  < 2e-16 ***
## campaign           -0.222311   0.029199  -7.614 2.66e-14 ***
## previous            0.698768   1.421252   0.492 0.622962
## poutcomefailure    -0.537251   1.422072  -0.378 0.705583
## poutcomeother      -0.307016   1.423769  -0.216 0.829271
## poutcomesuccess     1.787126   1.422801   1.256 0.209094
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 20268  on 30234  degrees of freedom
## Residual deviance: 13058  on 30194  degrees of freedom
## AIC: 13140
##
## Number of Fisher Scoring iterations: 6
```

We have checked the result again and we have excluded the variables that have p-value greater than 0.05 to apply the model again.

```
model <- glm(y ~ jobstudent + maritalmarried + educationprimary + balance + housing +
loan + contactcellular +
               contacttelephone + monthapr + monthaug + monthfeb + monthjan + monthju
l + monthjun + monthmar +
               monthmay + monthnov + duration + campaign,
               data = train, family = binomial)
```

```
summary(model)
```

```
##
## Call:
## glm(formula = y ~ jobstudent + maritalmarried + educationprimary +
##     balance + housing + loan + contactcellular + contacttelephone +
##     monthapr + monthaug + monthfeb + monthjan + monthjul + monthjun +
##     monthmar + monthmay + monthnov + duration + campaign, family = binomial,
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8598  -0.3794  -0.2378  -0.1293   3.7198
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -2.14714    0.11938 -17.986  < 2e-16 ***
## jobstudent        0.74967    0.11094   6.758 1.40e-11 ***
## maritalmarried   -0.14630    0.04640  -3.153  0.00161 **
## educationprimary -0.40724    0.07188  -5.666 1.46e-08 ***
## balance           0.15040    0.02066   7.279 3.37e-13 ***
## housing          -0.85003    0.05252 -16.184  < 2e-16 ***
## loan             -0.56159    0.07455  -7.533 4.95e-14 ***
## contactcellular   2.05088    0.08830  23.225  < 2e-16 ***
## contacttelephone  1.92801    0.12275  15.707  < 2e-16 ***
## monthapr         -1.37115    0.10282 -13.335  < 2e-16 ***
## monthaug         -1.96781    0.09637 -20.420  < 2e-16 ***
## monthfeb         -1.46850    0.10594 -13.861  < 2e-16 ***
## monthjan         -2.62838    0.15346 -17.128  < 2e-16 ***
## monthjul         -2.32390    0.10048 -23.129  < 2e-16 ***
## monthjun         -0.66900    0.10987  -6.089 1.14e-09 ***
## monthmar          0.45924    0.14678   3.129  0.00176 **
## monthmay         -1.67568    0.09628 -17.404  < 2e-16 ***
## monthnov         -2.09371    0.10770 -19.439  < 2e-16 ***
## duration          1.02166    0.01940  52.650  < 2e-16 ***
## campaign         -0.28277    0.02856  -9.901  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 20268  on 30234  degrees of freedom
## Residual deviance: 14077  on 30215  degrees of freedom
## AIC: 14117
##
## Number of Fisher Scoring iterations: 6
```

It can be understood that contactcellular, contacttelephone and duration are the variables which have greater affect on subscription.

```
predicttrains <- predict(model, train[-47], type = 'response')
predictions <- predict(model, test[-47], type = 'response')
```

Then we've looked at the performance metrics of the model.

```
predicted.classes <- ifelse(predicttrains > 0.5, "1", "0")
predicted.classes.test <- ifelse(predictions > 0.5, "1", "0")
```

```
mean(predicted.classes == train$y)
```

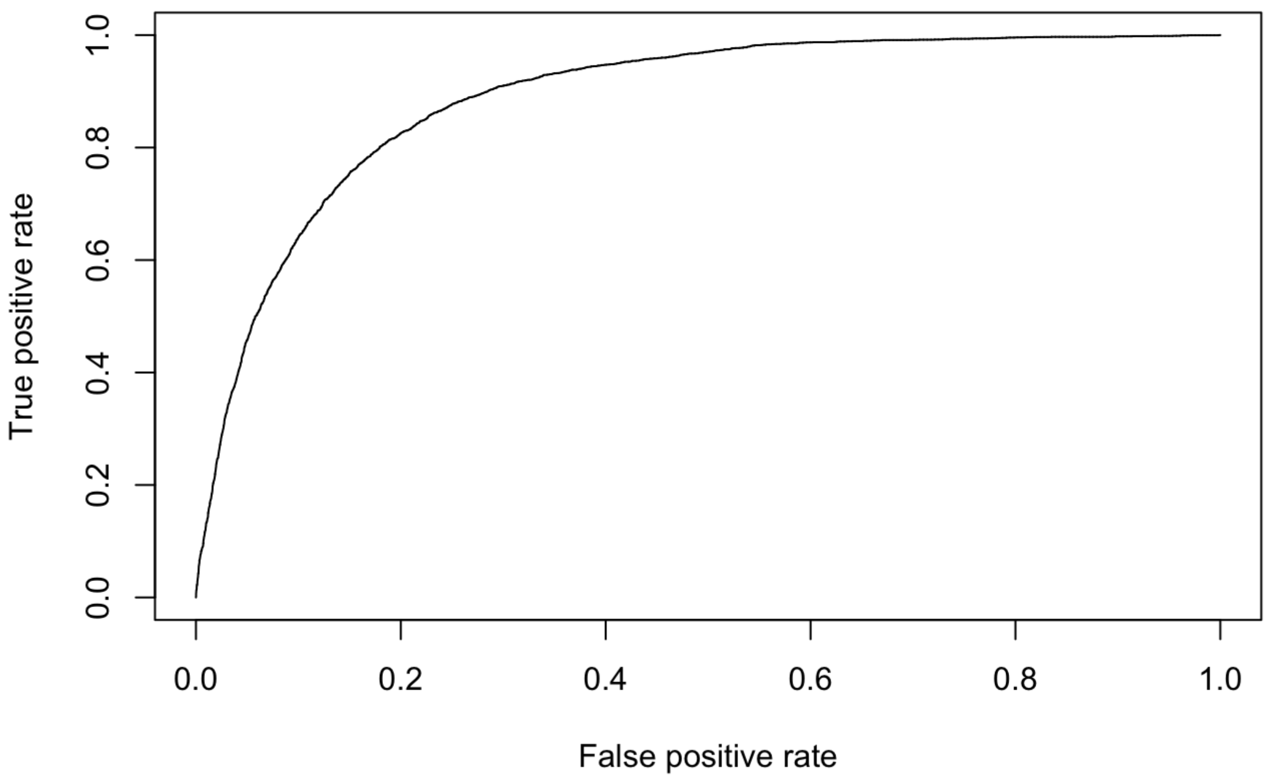
```
## [1] 0.9024971
```

```
mean(predicted.classes.test == test$y)
```

```
## [1] 0.9028673
```

```
train$y = factor(train$y)
test$y = factor(test$y)
```

```
pred = prediction(predicttrains, train$y)
perf = performance(pred, measure = "tpr", x.measure = "fpr")
plot(perf)
```



Confusion Matrix and Statistics:

```
train_cm <- factor(predicted.classes, levels = levels(as.factor(train[["y"]])))
confusionMatrix(train_cm, as.factor(train[["y"]]))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 26454  2331
##          1   617   833
##
##                Accuracy : 0.9025
##                  95% CI : (0.8991, 0.9058)
##     No Information Rate : 0.8954
##     P-Value [Acc > NIR] : 2.174e-05
##
##                   Kappa : 0.3161
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9772
##             Specificity : 0.2633
##          Pos Pred Value : 0.9190
##          Neg Pred Value : 0.5745
##              Prevalence : 0.8954
##          Detection Rate : 0.8749
##    Detection Prevalence : 0.9520
##       Balanced Accuracy : 0.6202
##
##        'Positive' Class : 0
##
```

```
test_cm <- factor(predicted.classes.test, levels = levels(as.factor(test[["y"]])))
confusionMatrix(test_cm, as.factor(test[["y"]]))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 8817  787
##          1  192  283
##
##                Accuracy : 0.9029
##                  95% CI : (0.8969, 0.9086)
##     No Information Rate : 0.8938
##     P-Value [Acc > NIR] : 0.001539
##
##                   Kappa : 0.3221
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9787
##             Specificity : 0.2645
##          Pos Pred Value : 0.9181
##          Neg Pred Value : 0.5958
##              Prevalence : 0.8938
##          Detection Rate : 0.8748
##    Detection Prevalence : 0.9529
##       Balanced Accuracy : 0.6216
##
##        'Positive' Class : 0
##
```

Accuracy score of the model is 0.903. While Sensitivity is 0.979, Specificity is 0.265.

Decision Tree

For better analysis, complexity parameter was selected as 0.003.

```
fit <- rpart(train$y ~ ., data = train, method="class", control = rpart.control(cp =
0.003))
```

```
summary(fit)
```

```
## Call:
## rpart(formula = train$y ~ ., data = train, method = "class",
##     control = rpart.control(cp = 0.003))
##   n= 30235
##
##            CP nsplit rel error    xerror       xstd
## 1 0.087547408      0 1.0000000 1.0000000 0.01682205
## 2 0.032237674      1 0.9124526 0.9124526 0.01615083
## 3 0.003792668      2 0.8802149 0.8814791 0.01590275
## 4 0.003000000     13 0.8305942 0.8669406 0.01578427
##
## Variable importance
##        duration poutcomesuccess             age        monthmar
##              43              40               6               5
##  contactunknown contactcellular         balance        monthmay
##               1               1               1               1
##
## Node number 1: 30235 observations,    complexity param=0.08754741
##   predicted class=0  expected loss=0.1046469  P(node) =1
##     class counts: 27071  3164
##    probabilities: 0.895 0.105
##   left son=2 (29214 obs) right son=3 (1021 obs)
##   Primary splits:
##       poutcomesuccess < 0.5         to the left,  improve=595.8966, (0 missing)
##       duration        < 0.9685519   to the left,  improve=480.6245, (0 missing)
##       previous        < 0.5         to the left,  improve=183.8680, (0 missing)
##       poutcomeunknown < 0.5         to the right, improve=183.4708, (0 missing)
##       age             < 1.870155    to the left,  improve=163.6994, (0 missing)
##   Surrogate splits:
##       age < 4.761374    to the left,  agree=0.966, adj=0.001, (0 split)
##
## Node number 2: 29214 observations,    complexity param=0.003792668
##   predicted class=0  expected loss=0.08608886  P(node) =0.9662312
##     class counts: 26699  2515
##    probabilities: 0.914 0.086
##   left son=4 (26080 obs) right son=5 (3134 obs)
##   Primary splits:
##       duration       < 1.301188    to the left,  improve=447.49100, (0 missing)
##       age            < 1.870155    to the left,  improve=102.93060, (0 missing)
##       monthmar       < 0.5         to the left,  improve= 81.97760, (0 missing)
##       contactunknown < 0.5         to the right, improve= 77.91345, (0 missing)
##       housing        < 0.5         to the right, improve= 71.90364, (0 missing)
##
## Node number 3: 1021 observations,    complexity param=0.03223767
##   predicted class=1  expected loss=0.3643487  P(node) =0.03376881
##     class counts:   372   649
##    probabilities: 0.364 0.636
##   left son=6 (260 obs) right son=7 (761 obs)
##   Primary splits:
##       duration < -0.3783536  to the left,  improve=76.808750, (0 missing)
##       housing  < 0.5         to the right, improve=17.318050, (0 missing)
##       monthmay < 0.5         to the right, improve= 9.153273, (0 missing)
##       age      < 0.1638615   to the left,  improve= 5.994868, (0 missing)
##       campaign < 0.6960037   to the right, improve= 3.922216, (0 missing)
##   Surrogate splits:
##       contactunknown < 0.5         to the right, agree=0.755, adj=0.038, (0 split)
##       campaign       < 3.086368    to the right, agree=0.749, adj=0.015, (0 split)
```

```
##       balance        < 3.821354    to the right, agree=0.746, adj=0.004, (0 split)
##
## Node number 4: 26080 observations,    complexity param=0.003792668
##   predicted class=0  expected loss=0.05575153  P(node) =0.8625765
##     class counts: 24626  1454
##    probabilities: 0.944 0.056
##   left son=8 (25545 obs) right son=9 (535 obs)
##   Primary splits:
##       age      < 1.870155    to the left,  improve=81.54771, (0 missing)
##       duration < -0.1166068  to the left,  improve=79.06853, (0 missing)
##       monthmar < 0.5         to the left,  improve=76.49488, (0 missing)
##       housing  < 0.5         to the right, improve=63.81565, (0 missing)
##       monthoct < 0.5         to the left,  improve=58.94779, (0 missing)
##
## Node number 5: 3134 observations,    complexity param=0.003792668
##   predicted class=0  expected loss=0.338545  P(node) =0.1036547
##     class counts:  2073  1061
##    probabilities: 0.661 0.339
##   left son=10 (1677 obs) right son=11 (1457 obs)
##   Primary splits:
##       duration         < 2.184584    to the left,  improve=63.82925, (0 missing)
##       contactunknown   < 0.5         to the right, improve=29.05860, (0 missing)
##       contactcellular  < 0.5         to the left,  improve=25.22372, (0 missing)
##       age              < 1.964949    to the left,  improve=16.63182, (0 missing)
##       housing          < 0.5         to the right, improve=13.12758, (0 missing)
##   Surrogate splits:
##       balance          < 1.53955     to the left,  agree=0.541, adj=0.012, (0 spli
t)
##       jobunemployed    < 0.5         to the left,  agree=0.539, adj=0.009, (0 spli
t)
##       jobself-employed < 0.5         to the left,  agree=0.537, adj=0.004, (0 spli
t)
##       campaign         < 1.891186    to the left,  agree=0.537, adj=0.004, (0 spli
t)
##       jobunknown       < 0.5         to the left,  agree=0.536, adj=0.001, (0 spli
t)
##
## Node number 6: 260 observations
##   predicted class=0  expected loss=0.3038462  P(node) =0.008599305
##     class counts:   181    79
##    probabilities: 0.696 0.304
##
## Node number 7: 761 observations
##   predicted class=1  expected loss=0.2509855  P(node) =0.02516951
##     class counts:   191   570
##    probabilities: 0.251 0.749
##
## Node number 8: 25545 observations,    complexity param=0.003792668
##   predicted class=0  expected loss=0.05002936  P(node) =0.8448818
##     class counts: 24267  1278
##    probabilities: 0.950 0.050
##   left son=16 (25335 obs) right son=17 (210 obs)
##   Primary splits:
##       monthmar       < 0.5         to the left,  improve=73.51071, (0 missing)
##       duration       < -0.1002476  to the left,  improve=62.97493, (0 missing)
##       monthoct       < 0.5         to the left,  improve=52.41519, (0 missing)
##       housing        < 0.5         to the right, improve=45.51140, (0 missing)
##       contactunknown < 0.5         to the right, improve=41.09764, (0 missing)
```

```
##
## Node number 9: 535 observations,    complexity param=0.003792668
##   predicted class=0  expected loss=0.328972  P(node) =0.01769472
##     class counts:   359   176
##    probabilities: 0.671 0.329
##   left son=18 (308 obs) right son=19 (227 obs)
##   Primary splits:
##       duration         < -0.138419   to the left,  improve=34.273500, (0 missing)
##       campaign         < 0.6960037   to the right, improve= 4.738662, (0 missing)
##       contactcellular  < 0.5         to the left,  improve= 4.075189, (0 missing)
##       balance          < 1.248798    to the left,  improve= 3.658906, (0 missing)
##       contacttelephone < 0.5         to the right, improve= 2.611594, (0 missing)
##   Surrogate splits:
##       balance          < 3.288309    to the left,  agree=0.585, adj=0.022, (0 spli
t)
##       educationunknown < 0.5         to the left,  agree=0.583, adj=0.018, (0 spli
t)
##       monthjun         < 0.5         to the left,  agree=0.583, adj=0.018, (0 spli
t)
##       jobadmin.        < 0.5         to the left,  agree=0.581, adj=0.013, (0 spli
t)
##       jobunknown       < 0.5         to the left,  agree=0.581, adj=0.013, (0 spli
t)
##
## Node number 10: 1677 observations
##   predicted class=0  expected loss=0.2444842  P(node) =0.05546552
##     class counts:  1267   410
##    probabilities: 0.756 0.244
##
## Node number 11: 1457 observations,    complexity param=0.003792668
##   predicted class=0  expected loss=0.4468085  P(node) =0.04818918
##     class counts:   806   651
##    probabilities: 0.553 0.447
##   left son=22 (477 obs) right son=23 (980 obs)
##   Primary splits:
##       contactcellular < 0.5         to the left,  improve=16.938760, (0 missing)
##       contactunknown  < 0.5         to the right, improve=15.727960, (0 missing)
##       duration        < 3.231571    to the left,  improve=10.027060, (0 missing)
##       maritalmarried  < 0.5         to the right, improve= 8.179196, (0 missing)
##       age             < 1.775361    to the left,  improve= 7.558276, (0 missing)
##   Surrogate splits:
##       contactunknown   < 0.5         to the right, agree=0.962, adj=0.885, (0 spli
t)
##       monthjun         < 0.5         to the right, agree=0.747, adj=0.229, (0 spli
t)
##       monthmay         < 0.5         to the right, agree=0.738, adj=0.199, (0 spli
t)
##       contacttelephone < 0.5         to the right, agree=0.710, adj=0.115, (0 spli
t)
##       age              < 3.955624    to the right, agree=0.675, adj=0.006, (0 spli
t)
##
## Node number 16: 25335 observations
##   predicted class=0  expected loss=0.04657588  P(node) =0.8379362
##     class counts: 24155  1180
##    probabilities: 0.953 0.047
##
## Node number 17: 210 observations,    complexity param=0.003792668
```

```
##   predicted class=0  expected loss=0.4666667  P(node) =0.006945593
##     class counts:   112    98
##    probabilities: 0.533 0.467
##   left son=34 (116 obs) right son=35 (94 obs)
##   Primary splits:
##       duration         < -0.3183699  to the left,  improve=17.202810, (0 missing)
##       balance          < -0.1403482  to the right, improve= 3.368845, (0 missing)
##       contacttelephone < 0.5         to the right, improve= 2.712170, (0 missing)
##       poutcomefailure  < 0.5         to the right, improve= 2.454472, (0 missing)
##       previous         < 0.5         to the right, improve= 2.396463, (0 missing)
##   Surrogate splits:
##       poutcomeother   < 0.5         to the left,  agree=0.586, adj=0.074, (0 spli
t)
##       age             < 1.680567    to the left,  agree=0.576, adj=0.053, (0 spli
t)
##       previous        < 0.5         to the left,  agree=0.576, adj=0.053, (0 spli
t)
##       poutcomeunknown < 0.5         to the right, agree=0.576, adj=0.053, (0 spli
t)
##       jobservices     < 0.5         to the left,  agree=0.567, adj=0.032, (0 spli
t)
##
## Node number 18: 308 observations
##   predicted class=0  expected loss=0.1753247  P(node) =0.01018687
##     class counts:   254    54
##    probabilities: 0.825 0.175
##
## Node number 19: 227 observations,    complexity param=0.003792668
##   predicted class=1  expected loss=0.4625551  P(node) =0.007507855
##     class counts:   105   122
##    probabilities: 0.463 0.537
##   left son=38 (62 obs) right son=39 (165 obs)
##   Primary splits:
##       balance         < -0.381791   to the left,  improve=3.856202, (0 missing)
##       age             < 1.964949    to the left,  improve=3.315049, (0 missing)
##       duration        < -0.08388842 to the right, improve=2.315016, (0 missing)
##       poutcomefailure < 0.5         to the right, improve=2.199770, (0 missing)
##       previous        < 0.5         to the right, improve=1.775913, (0 missing)
##   Surrogate splits:
##       jobself-employed < 0.5         to the right, agree=0.736, adj=0.032, (0 spli
t)
##       contactunknown   < 0.5         to the right, agree=0.736, adj=0.032, (0 spli
t)
##       age              < 3.671242    to the right, agree=0.731, adj=0.016, (0 spli
t)
##       jobblue-collar   < 0.5         to the right, agree=0.731, adj=0.016, (0 spli
t)
##
## Node number 22: 477 observations
##   predicted class=0  expected loss=0.3375262  P(node) =0.01577642
##     class counts:   316   161
##    probabilities: 0.662 0.338
##
## Node number 23: 980 observations,    complexity param=0.003792668
##   predicted class=0  expected loss=0.5  P(node) =0.03241277
##     class counts:   490   490
##    probabilities: 0.500 0.500
##   left son=46 (789 obs) right son=47 (191 obs)
```

```
##   Primary splits:
##       monthmay < 0.5         to the left,  improve=6.012051, (0 missing)
##       monthjan < 0.5         to the right, improve=5.612873, (0 missing)
##       monthapr < 0.5         to the right, improve=5.397010, (0 missing)
##       duration < 3.155228    to the left,  improve=5.116834, (0 missing)
##       balance  < 0.6303133   to the left,  improve=4.815252, (0 missing)
##   Surrogate splits:
##       age < -1.73202    to the right, agree=0.807, adj=0.01, (0 split)
##
## Node number 34: 116 observations
##   predicted class=0  expected loss=0.2844828  P(node) =0.003836613
##     class counts:    83    33
##    probabilities: 0.716 0.284
##
## Node number 35: 94 observations
##   predicted class=1  expected loss=0.3085106  P(node) =0.00310898
##     class counts:    29    65
##    probabilities: 0.309 0.691
##
## Node number 38: 62 observations
##   predicted class=0  expected loss=0.3870968  P(node) =0.002050604
##     class counts:    38    24
##    probabilities: 0.613 0.387
##
## Node number 39: 165 observations
##   predicted class=1  expected loss=0.4060606  P(node) =0.005457252
##     class counts:    67    98
##    probabilities: 0.406 0.594
##
## Node number 46: 789 observations,    complexity param=0.003792668
##   predicted class=0  expected loss=0.4727503  P(node) =0.02609558
##     class counts:   416   373
##    probabilities: 0.527 0.473
##   left son=92 (423 obs) right son=93 (366 obs)
##   Primary splits:
##       balance  < -0.1735041  to the left,  improve=6.876107, (0 missing)
##       age      < 1.775361    to the left,  improve=5.864224, (0 missing)
##       monthjan < 0.5         to the right, improve=4.529842, (0 missing)
##       duration < 2.473596    to the left,  improve=4.505834, (0 missing)
##       housing  < 0.5         to the right, improve=4.281632, (0 missing)
##   Surrogate splits:
##       age        < 0.2586556   to the left,  agree=0.598, adj=0.134, (0 split)
##       monthjul   < 0.5         to the right, agree=0.561, adj=0.055, (0 split)
##       monthnov   < 0.5         to the left,  agree=0.559, adj=0.049, (0 split)
##       duration   < 3.52331     to the left,  agree=0.556, adj=0.044, (0 split)
##       jobretired < 0.5         to the left,  agree=0.553, adj=0.036, (0 split)
##
## Node number 47: 191 observations
##   predicted class=1  expected loss=0.3874346  P(node) =0.006317182
##     class counts:    74   117
##    probabilities: 0.387 0.613
##
## Node number 92: 423 observations
##   predicted class=0  expected loss=0.4113475  P(node) =0.01399041
##     class counts:   249   174
##    probabilities: 0.589 0.411
##
## Node number 93: 366 observations,    complexity param=0.003792668
```

```
##   predicted class=1  expected loss=0.4562842  P(node) =0.01210518
##     class counts:   167   199
##    probabilities: 0.456 0.544
##   left son=186 (151 obs) right son=187 (215 obs)
##   Primary splits:
##       housing        < 0.5         to the right, improve=4.483335, (0 missing)
##       duration       < 3.340632    to the left,  improve=3.108612, (0 missing)
##       monthaug       < 0.5         to the left,  improve=3.071396, (0 missing)
##       jobblue-collar < 0.5         to the right, improve=2.985329, (0 missing)
##       age            < 1.964949    to the left,  improve=2.779128, (0 missing)
##   Surrogate splits:
##       monthapr        < 0.5         to the right, agree=0.678, adj=0.219, (0 spli
t)
##       jobblue-collar  < 0.5         to the right, agree=0.631, adj=0.106, (0 spli
t)
##       previous        < 0.5         to the right, agree=0.609, adj=0.053, (0 spli
t)
##       poutcomeunknown < 0.5         to the left,  agree=0.609, adj=0.053, (0 spli
t)
##       poutcomefailure < 0.5         to the right, agree=0.604, adj=0.040, (0 spli
t)
##
## Node number 186: 151 observations
##   predicted class=0  expected loss=0.4503311  P(node) =0.004994212
##     class counts:    83    68
##    probabilities: 0.550 0.450
##
## Node number 187: 215 observations
##   predicted class=1  expected loss=0.3906977  P(node) =0.007110964
##     class counts:    84   131
##    probabilities: 0.391 0.609
```

It can be understood that ‘duration’ and ‘poutcomesuccess’ have the highest variable importance percentages. The longer the last contact duration is, the higher the probability of customer subscripting for the term deposit. Similarly, success of the previous marketing campaign has a significant effect on subscription.

```
predicted_train = predict(fit, train[-47], type = "class")
predicted_test = predict(fit, test[-47], type = "class")
```

Confusion Matrix:

```
table = table(test$y, predicted_test)
table
```

```
##    predicted_test
##        0    1
##   0 8848  161
##   1  752  318
```

```
accuracy = sum(diag(table)) / sum(table)
accuracy
```

```
## [1] 0.9094156
```

Accuracy score of the decision tree model is 0.91.

The visualization of the decision tree as below:

```
rpart.plot(fit, extra=106)
```