Marketing Campaign

Sri Harika Cherukuri

27/11/2021

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v stringr 1.4.0
## v tidyr   1.1.3     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
```

```
data = read_delim("bank-additional-full.csv", delim=";")
```

```
## Rows: 41188 Columns: 21

## -- Column specification --------------------------------------------------------
## Delimiter: ";"
## chr (11): job, marital, education, default, housing, loan, contact, month, d...
## dbl (10): age, duration, campaign, pdays, previous, emp.var.rate, cons.price...

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(data,10)
```

```
## # A tibble: 10 x 21
##      age job         marital education default housing loan  contact month day_of_week
##    <dbl> <chr>       <chr>   <chr>     <chr>   <chr>   <chr> <chr>   <chr> <chr>
##  1    56 housemaid   married basic.4y  no      no      no    teleph~ may   mon
##  2    57 services    married high.sch~ unknown no      no    teleph~ may   mon
##  3    37 services    married high.sch~ no      yes     no    teleph~ may   mon
##  4    40 admin.      married basic.6y  no      no      no    teleph~ may   mon
##  5    56 services    married high.sch~ no      no      yes   teleph~ may   mon
##  6    45 services    married basic.9y  unknown no      no    teleph~ may   mon
##  7    59 admin.      married professi~ no      no      no    teleph~ may   mon
##  8    41 blue-collar married unknown   unknown no      no    teleph~ may   mon
##  9    24 technician  single  professi~ no      yes     no    teleph~ may   mon
## 10    25 services    single  high.sch~ no      yes     no    teleph~ may   mon
## # ... with 11 more variables: duration <dbl>, campaign <dbl>, pdays <dbl>,
## #   previous <dbl>, poutcome <chr>, emp.var.rate <dbl>, cons.price.idx <dbl>,
## #   cons.conf.idx <dbl>, euribor3m <dbl>, nr.employed <dbl>, y <chr>
```

```
summary(data)
```

```
##       age            job              marital           education
##  Min.   :17.00   Length:41188       Length:41188       Length:41188
##  1st Qu.:32.00   Class :character   Class :character   Class :character
##  Median :38.00   Mode  :character   Mode  :character   Mode  :character
##  Mean   :40.02
##  3rd Qu.:47.00
##  Max.   :98.00
##    default            housing              loan             contact
##  Length:41188       Length:41188       Length:41188       Length:41188
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##     month           day_of_week           duration         campaign
##  Length:41188       Length:41188       Min.   :   0.0   Min.   : 1.000
##  Class :character   Class :character   1st Qu.: 102.0   1st Qu.: 1.000
##  Mode  :character   Mode  :character   Median : 180.0   Median : 2.000
##                                        Mean   : 258.3   Mean   : 2.568
##                                        3rd Qu.: 319.0   3rd Qu.: 3.000
##                                        Max.   :4918.0   Max.   :56.000
##      pdays          previous       poutcome          emp.var.rate
##  Min.   :  0.0   Min.   :0.000   Length:41188       Min.   :-3.40000
##  1st Qu.:999.0   1st Qu.:0.000   Class :character   1st Qu.:-1.80000
##  Median :999.0   Median :0.000   Mode  :character   Median : 1.10000
##  Mean   :962.5   Mean   :0.173                      Mean   : 0.08189
##  3rd Qu.:999.0   3rd Qu.:0.000                      3rd Qu.: 1.40000
##  Max.   :999.0   Max.   :7.000                      Max.   : 1.40000
##  cons.price.idx  cons.conf.idx     euribor3m      nr.employed
##  Min.   :92.20   Min.   :-50.8   Min.   :0.634   Min.   :4964
##  1st Qu.:93.08   1st Qu.:-42.7   1st Qu.:1.344   1st Qu.:5099
```

```
##  Median :93.75   Median :-41.8   Median :4.857   Median :5191
##  Mean   :93.58   Mean   :-40.5   Mean   :3.621   Mean   :5167
##  3rd Qu.:93.99   3rd Qu.:-36.4   3rd Qu.:4.961   3rd Qu.:5228
##  Max.   :94.77   Max.   :-26.9   Max.   :5.045   Max.   :5228
##       y
##  Length:41188
##  Class :character
##  Mode  :character
##
##
##
```

```
sum(is.na(data))
```

```
## [1] 0
```

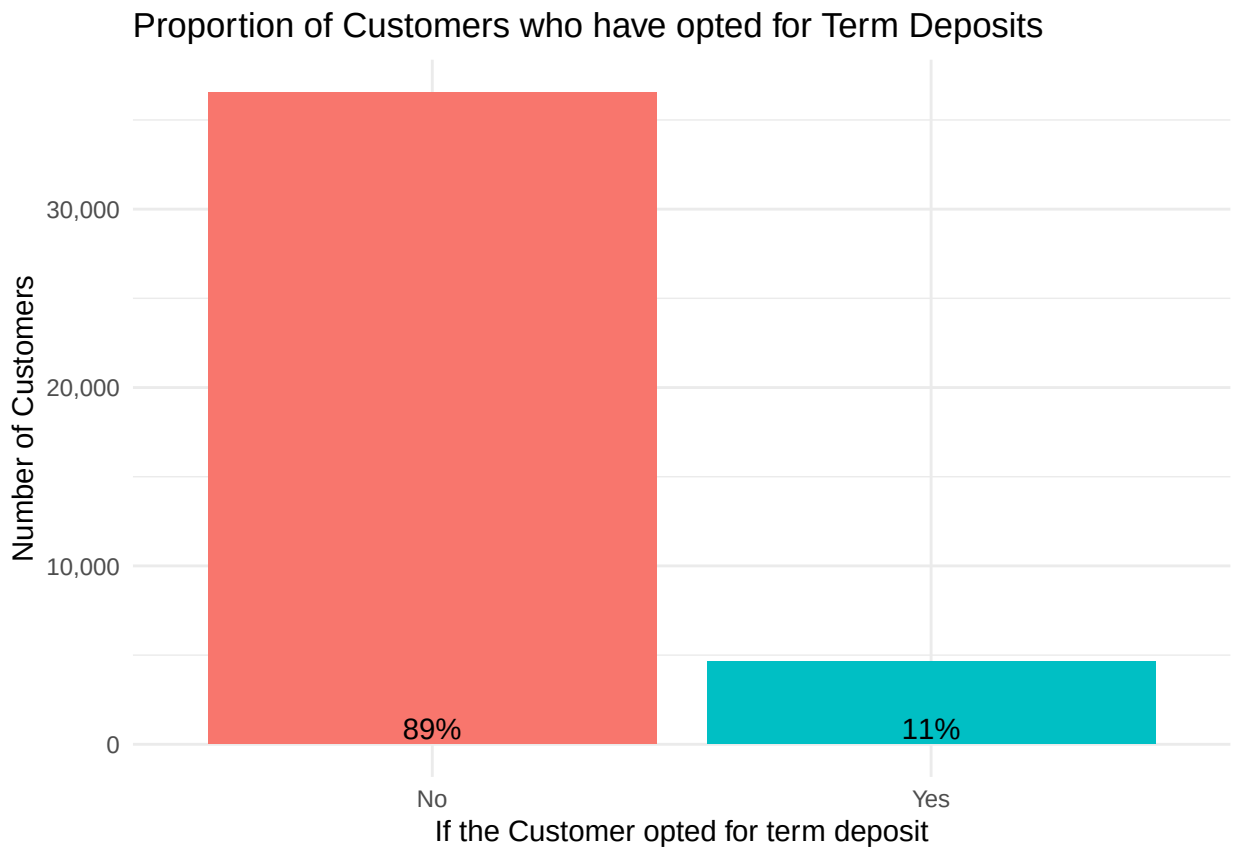
```
any(is.null(data))
```

```
## [1] FALSE
```

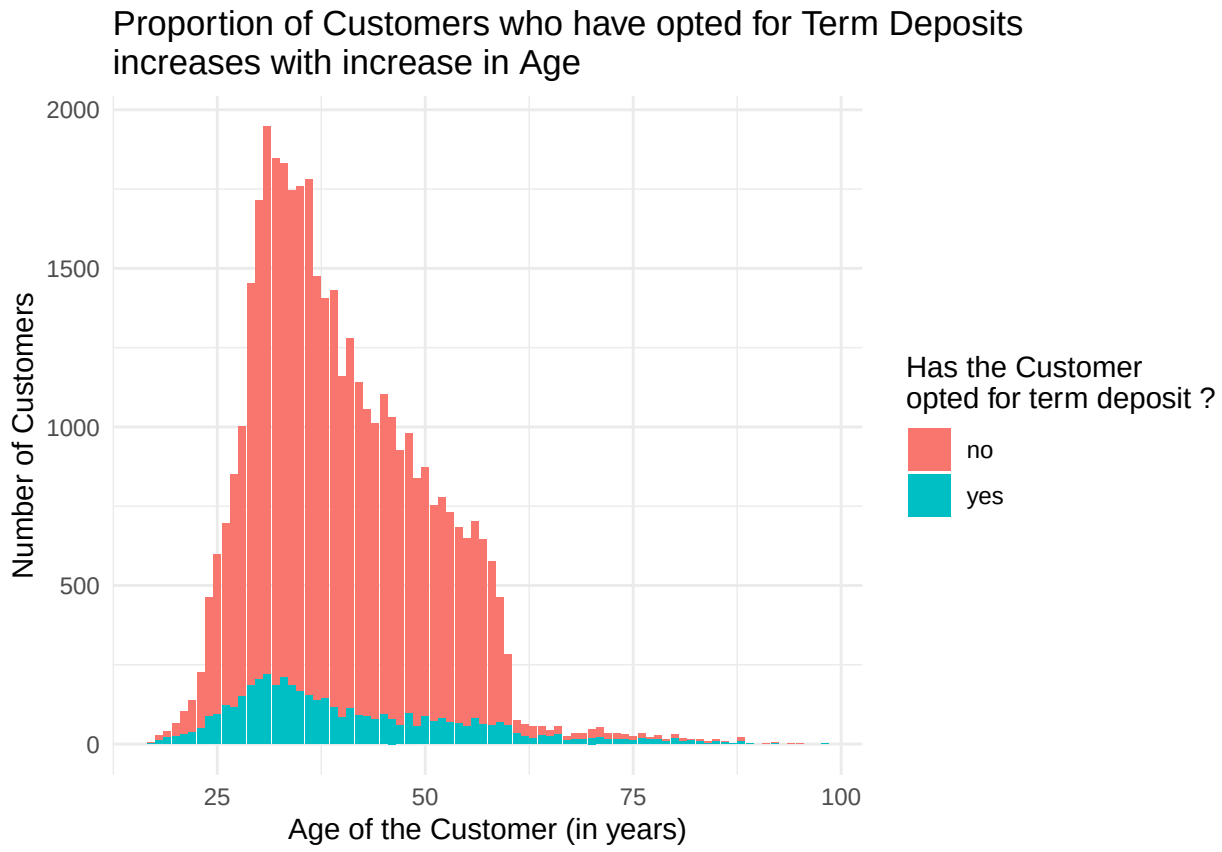
```
any(is.na(data))
```

```
## [1] FALSE
```

```
ggplot(data %>%
         count(y),
       aes(y,n,fill=y))+
  geom_bar(stat="identity")+
  labs(title = "Proportion of Customers who have opted for Term Deposits ", x = "If the Customer opted
  theme_minimal()+
  scale_x_discrete(labels = c("No","Yes"))+
  scale_y_continuous(labels = scales::number_format(big.mark = ','))+
  geom_text(aes(y = ((n)/sum(n)), label = scales::percent((n)/sum(n))),vjust = -0.25) +
  theme(legend.position = "none")
```



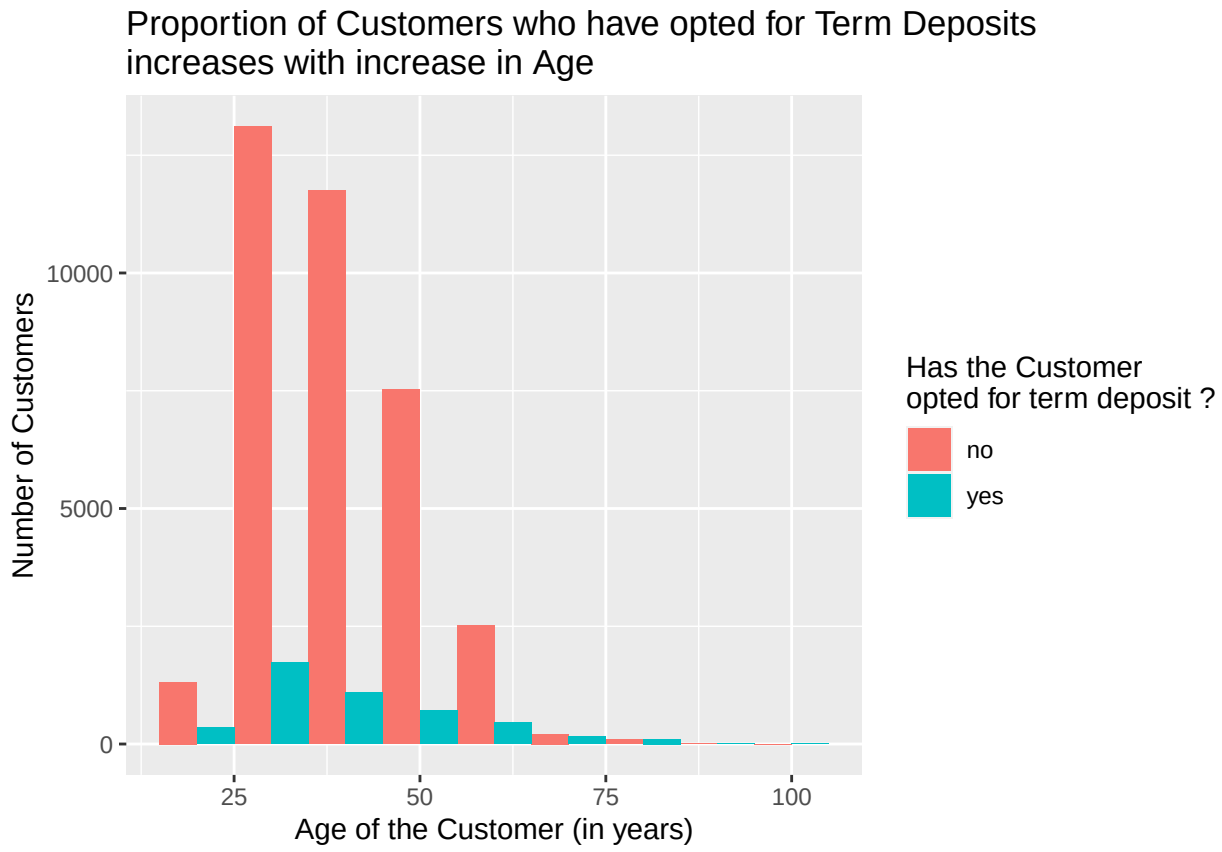

```
ggplot(data %>%
         count(age,y)%>%
         mutate(pct=n/sum(n)),
       aes(age,n,fill=y))+

  geom_bar(stat="identity")+
  labs(title = "Proportion of Customers who have opted for Term Deposits \nincreases with increase in Ag
  guides(fill=guide_legend(title="Has the Customer \nopted for term deposit ?"))+
  theme_minimal()
```




```
data %>% count(age)
```

```
## # A tibble: 78 x 2
##      age     n
##    <dbl> <int>
##  1    17     5
##  2    18    28
##  3    19    42
##  4    20    65
##  5    21   102
##  6    22   137
##  7    23   226
##  8    24   463
##  9    25   598
## 10    26   698
## # ... with 68 more rows
```

```
ggplot(data)+
  geom_histogram(mapping = aes(x = age,fill=y),binwidth = 10,position="dodge")+
  labs(title = "Proportion of Customers who have opted for Term Deposits \nincreases with increase in Ag
  guides(fill=guide_legend(title="Has the Customer \nopted for term deposit ?"))
```




```
nrow(data%>%filter(age=="unknown"))/nrow(data)*100
```

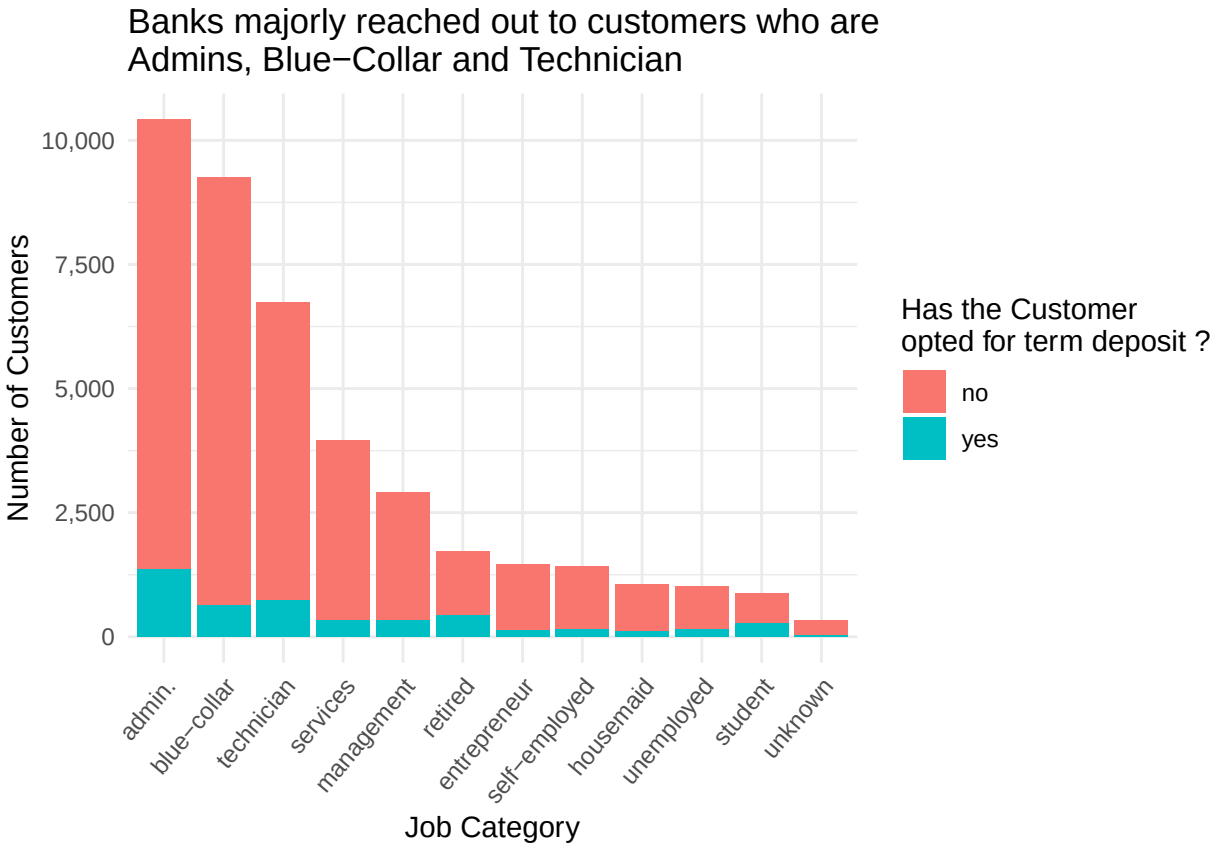
```
## [1] 0
```

```
data %>% count(job)
```

```
## # A tibble: 12 x 2
##    job             n
##    <chr>       <int>
##  1 admin.      10422
##  2 blue-collar  9254
##  3 entrepreneur 1456
##  4 housemaid    1060
##  5 management   2924
##  6 retired      1720
##  7 self-employed 1421
##  8 services     3969
##  9 student       875
## 10 technician   6743
## 11 unemployed   1014
## 12 unknown       330
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(job),
```

```
          fill = y)) +
geom_bar()+
labs(title="Banks majorly reached out to customers who are \nAdmins, Blue-Collar and Technician ",
     x="Job Category",
     y="Number of Customers")+
theme_minimal()+
scale_y_continuous(labels = scales::number_format(big.mark = ','))+
theme(axis.text.x=element_text(angle=50, hjust=1))+
guides(fill=guide_legend(title="Has the Customer \nopted for term deposit ?"))
```




```
nrow(data%>%filter(job=="unknown"))/nrow(data)*100
```

```
## [1] 0.8012042
```

Most people contacted during the campaigns are from the “admin” job category and they are the ones who are highest in number of people who agreed for a term deposit.

we can replace job values with retired where age is greater than or equal to 60.

Rest of the rows with unknown job values can be dropped off.

```r
#unknown job values before imputation
data %>% filter(job=="unknown")
```

```
## # A tibble: 330 x 21
##      age job     marital  education default housing loan  contact month day_of_week
##    <dbl> <chr>   <chr>    <chr>     <chr>   <chr>   <chr> <chr>   <chr> <chr>
##  1    55 unknown married  universi~ unknown unknown unkn~ teleph~ may   mon
##  2    55 unknown married  basic.4y  unknown yes     no    teleph~ may   mon
##  3    57 unknown married  unknown   unknown no      no    teleph~ may   mon
##  4    57 unknown married  unknown   unknown yes     no    teleph~ may   mon
##  5    38 unknown divorced high.sch~ unknown yes     no    teleph~ may   mon
##  6    38 unknown married  unknown   unknown no      no    teleph~ may   mon
##  7    43 unknown married  unknown   no      yes     no    teleph~ may   mon
##  8    57 unknown married  unknown   unknown yes     no    teleph~ may   mon
##  9    28 unknown single   unknown   unknown yes     yes   teleph~ may   tue
## 10    50 unknown married  unknown   unknown yes     no    teleph~ may   tue
## # ... with 320 more rows, and 11 more variables: duration <dbl>,
## #   campaign <dbl>, pdays <dbl>, previous <dbl>, poutcome <chr>,
## #   emp.var.rate <dbl>, cons.price.idx <dbl>, cons.conf.idx <dbl>,
## #   euribor3m <dbl>, nr.employed <dbl>, y <chr>
```

```r
data$job[data$job=="admin."] <- "admin"
data$job[data$job=="unknown" & data$age>=60] <- "retired"
#unknown job values after imputation
data %>% filter(job=="unknown")
```

```
## # A tibble: 301 x 21
##      age job     marital  education default housing loan  contact month day_of_week
##    <dbl> <chr>   <chr>    <chr>     <chr>   <chr>   <chr> <chr>   <chr> <chr>
##  1    55 unknown married  universi~ unknown unknown unkn~ teleph~ may   mon
##  2    55 unknown married  basic.4y  unknown yes     no    teleph~ may   mon
##  3    57 unknown married  unknown   unknown no      no    teleph~ may   mon
##  4    57 unknown married  unknown   unknown yes     no    teleph~ may   mon
##  5    38 unknown divorced high.sch~ unknown yes     no    teleph~ may   mon
##  6    38 unknown married  unknown   unknown no      no    teleph~ may   mon
##  7    43 unknown married  unknown   no      yes     no    teleph~ may   mon
##  8    57 unknown married  unknown   unknown yes     no    teleph~ may   mon
##  9    28 unknown single   unknown   unknown yes     yes   teleph~ may   tue
## 10    50 unknown married  unknown   unknown yes     no    teleph~ may   tue
## # ... with 291 more rows, and 11 more variables: duration <dbl>,
## #   campaign <dbl>, pdays <dbl>, previous <dbl>, poutcome <chr>,
## #   emp.var.rate <dbl>, cons.price.idx <dbl>, cons.conf.idx <dbl>,
## #   euribor3m <dbl>, nr.employed <dbl>, y <chr>
```

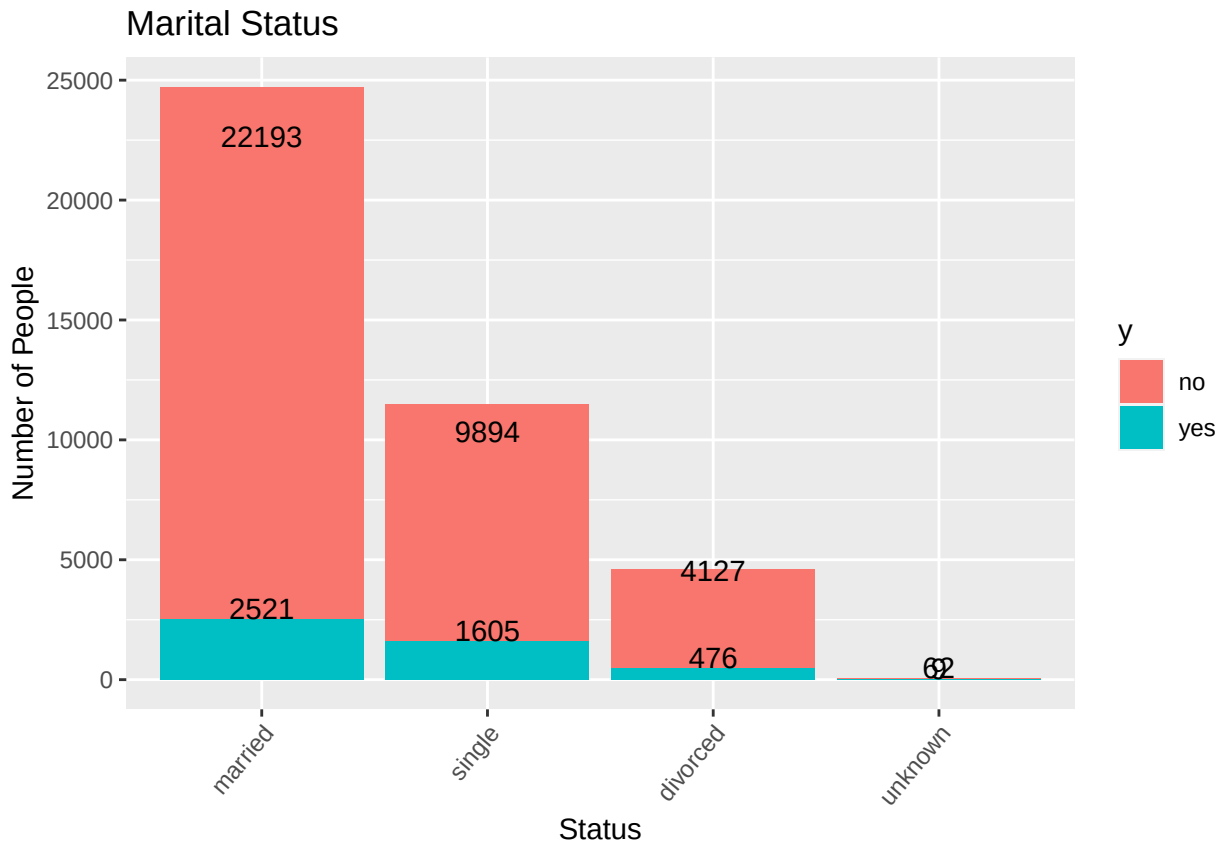
```r
data = data %>% filter(job!="unknown")
```

```r
data %>% count(marital)
```

```
## # A tibble: 4 x 2
##   marital      n
##   <chr>    <int>
```

```
## 1 divorced  4603
## 2 married  24714
## 3 single   11499
## 4 unknown     71
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(marital), fill=y)) +
  geom_bar()+
  stat_count(aes(label=..count..),
             vjust=0,
             geom="text",
             position="identity")+
  theme(axis.text.x=element_text(angle=50, hjust=1))+
  labs(title="Marital Status",
       x="Status",
       y="Number of People")
```




```
nrow(data%>%filter(marital=="unknown"))/nrow(data)*100
```

```
## [1] 0.1736493
```

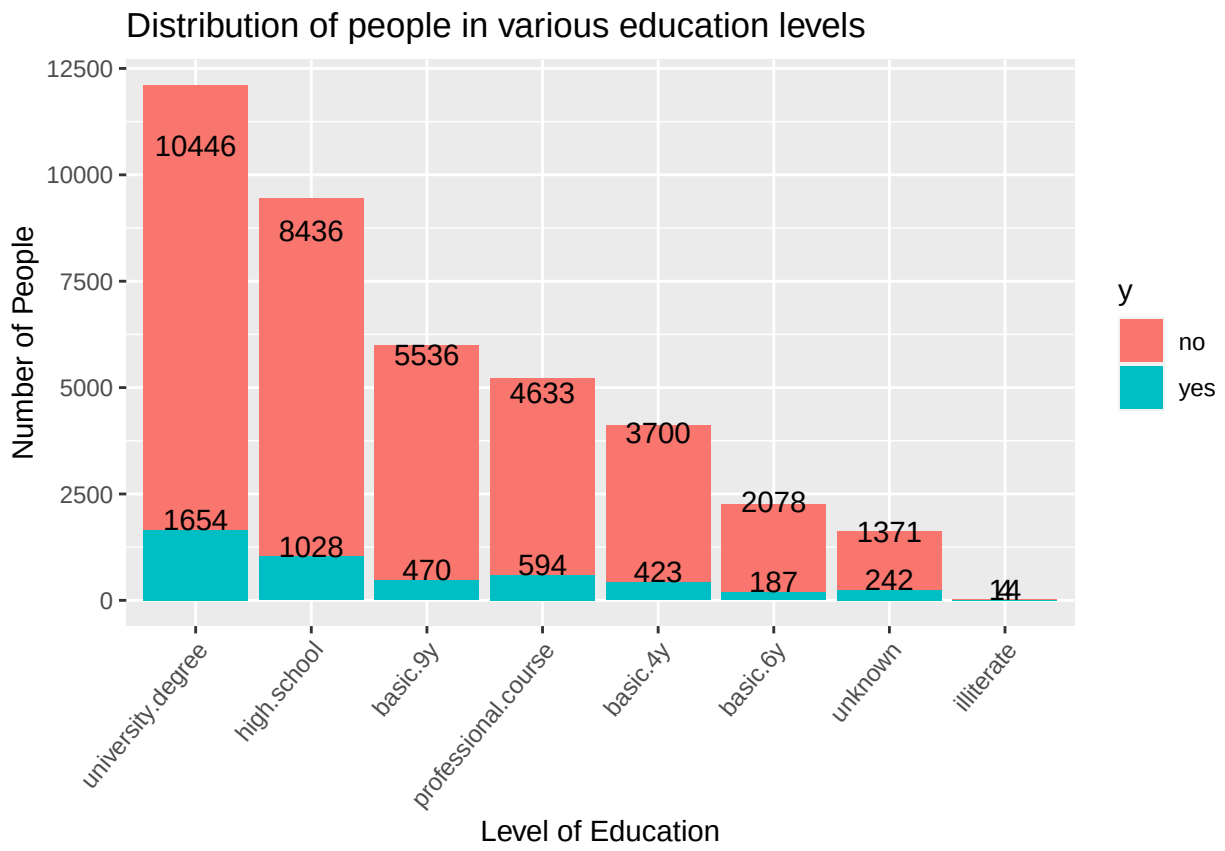
Dropping off the rows with unknown Marital Status values:

```
data = data %>% filter(marital!="unknown")
```

```
data %>% count(education)
```

```
## # A tibble: 8 x 2
##   education               n
##   <chr>               <int>
## 1 basic.4y             4123
## 2 basic.6y             2265
## 3 basic.9y             6006
## 4 high.school          9464
## 5 illiterate             18
## 6 professional.course  5227
## 7 university.degree   12100
## 8 unknown              1613
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(education), fill=y)) +
  geom_bar()+
  stat_count(aes(label=..count..),
             vjust=0,
             geom="text",
             position="identity")+
  theme(axis.text.x=element_text(angle=50, hjust=1))+
  labs(title="Distribution of people in various education levels",
       x="Level of Education",
       y="Number of People")
```

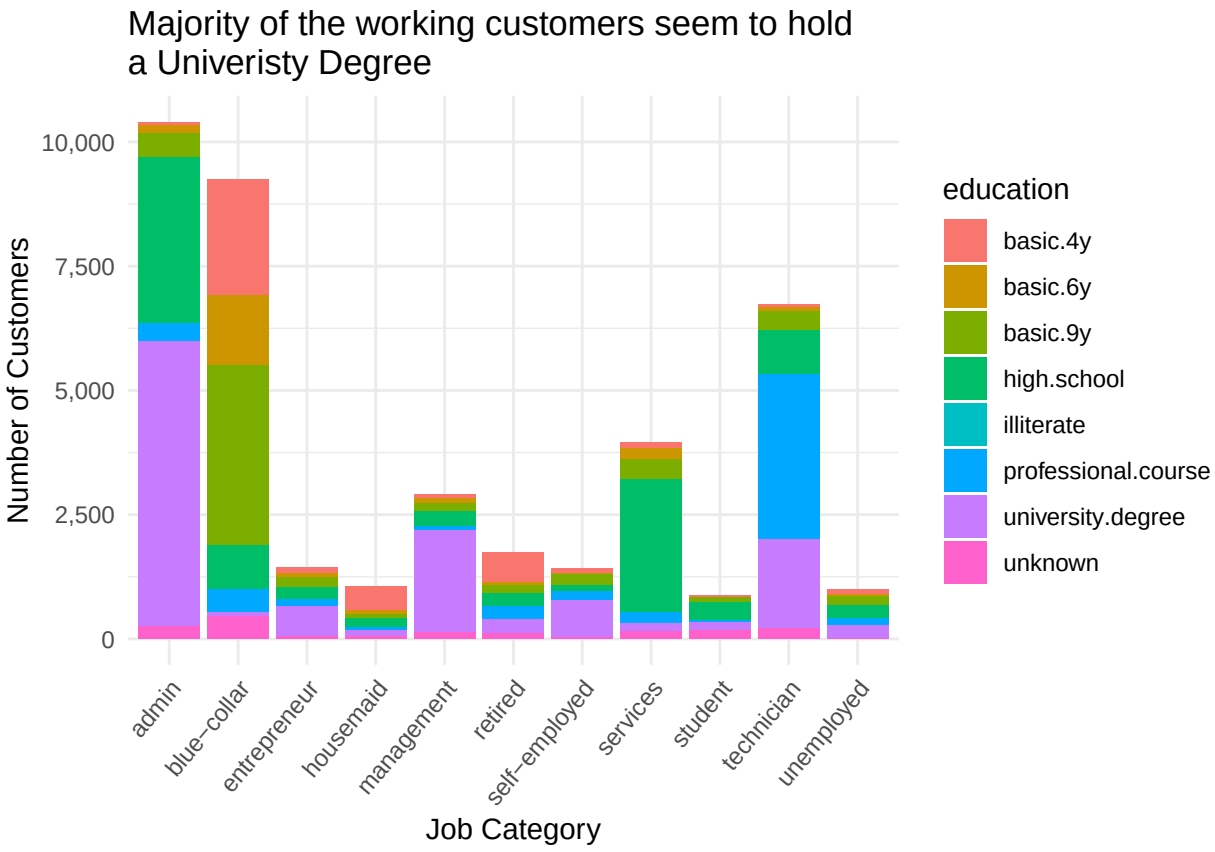



```
nrow(data%>%filter(education=="unknown"))/nrow(data)*100
```

```
## [1] 3.951882
```

Around 4% unknown values for education.
Hypothesis is that the job will be related to the level of education, so we can fill out the education level with the help of job positions.

```
ggplot(data,
       aes(x = job,
           fill = education)) +
  geom_bar(position = "stack")+
   labs(title = "Majority of the working customers seem to hold \na Univeristy Degree", x = "Job Categor
  theme_minimal()+
  scale_y_continuous(labels = scales::number_format(big.mark = ','))+
  theme(axis.text.x=element_text(angle=50, hjust=1))
```



Most occurring level of education in various jobs :

admin -> university.degree
blue-collar -> basic.9y
housemaid -> basic.4y
management -> university.degree
services -> high.school
technician -> professional.course

We can insert some education level values according to jobs:

```
#unknown education before imputation

data %>% filter(education=="unknown")
```

```
## # A tibble: 1,613 x 21
##      age job         marital education default housing loan  contact month day_of_week
##    <dbl> <chr>       <chr>   <chr>     <chr>   <chr>   <chr> <chr>   <chr> <chr>
##  1    41 blue-collar married unknown   unknown no      no    teleph~ may   mon
##  2    41 blue-collar married unknown   unknown no      no    teleph~ may   mon
##  3    59 technician  married unknown   no      yes     no    teleph~ may   mon
##  4    46 admin       married unknown   no      no      no    teleph~ may   mon
##  5    59 technician  married unknown   no      yes     no    teleph~ may   mon
##  6    49 blue-collar married unknown   no      no      no    teleph~ may   mon
##  7    33 admin       married unknown   no      yes     no    teleph~ may   mon
##  8    55 management  married unknown   unknown yes     no    teleph~ may   mon
```

```
##  9    60 admin       married unknown   unknown no      yes   teleph~ may   mon
## 10    54 services    married unknown   no      yes     no    teleph~ may   mon
## # ... with 1,603 more rows, and 11 more variables: duration <dbl>,
## #   campaign <dbl>, pdays <dbl>, previous <dbl>, poutcome <chr>,
## #   emp.var.rate <dbl>, cons.price.idx <dbl>, cons.conf.idx <dbl>,
## #   euribor3m <dbl>, nr.employed <dbl>, y <chr>
```

```
data$education[data$education=="unknown" & data$job=="admin"] <- "university.degree"

data$education[data$education=="unknown" & data$job=="blue-collar"] <- "basic.9y"

data$education[data$education=="unknown" & data$job=="housemaid"] <- "basic.4y"

data$education[data$education=="unknown" & data$job=="management"] <- "university.degree"

data$education[data$education=="unknown" & data$job=="services"] <- "high.school"

data$education[data$education=="unknown" & data$job=="technician"] <- "professional.course"


#unknoum education after imputation
data %>% filter(education=="unknown")
```

```
## # A tibble: 387 x 21
##      age job   marital education default housing loan  contact month day_of_week
##    <dbl> <chr> <chr>   <chr>     <chr>   <chr>   <chr> <chr>   <chr> <chr>
##  1    42 entr~ married unknown   unknown yes     no    teleph~ may   mon
##  2    56 entr~ married unknown   unknown yes     no    teleph~ may   mon
##  3    56 entr~ married unknown   unknown no      no    teleph~ may   mon
##  4    57 reti~ married unknown   unknown no      no    teleph~ may   mon
##  5    30 stud~ single  unknown   unknown no      no    teleph~ may   tue
##  6    60 reti~ single  unknown   unknown yes     no    teleph~ may   tue
##  7    55 self~ married unknown   unknown no      no    teleph~ may   tue
##  8    59 reti~ married unknown   no      yes     no    teleph~ may   thu
##  9    36 entr~ married unknown   unknown yes     no    teleph~ may   fri
## 10    26 entr~ married unknown   no      no      no    teleph~ may   fri
## # ... with 377 more rows, and 11 more variables: duration <dbl>,
## #   campaign <dbl>, pdays <dbl>, previous <dbl>, poutcome <chr>,
## #   emp.var.rate <dbl>, cons.price.idx <dbl>, cons.conf.idx <dbl>,
## #   euribor3m <dbl>, nr.employed <dbl>, y <chr>
```

1613-387 = 1226 values imputed, 387 will be dropped from the education column.

Now we can drop off the remaining rows with unknown education values:

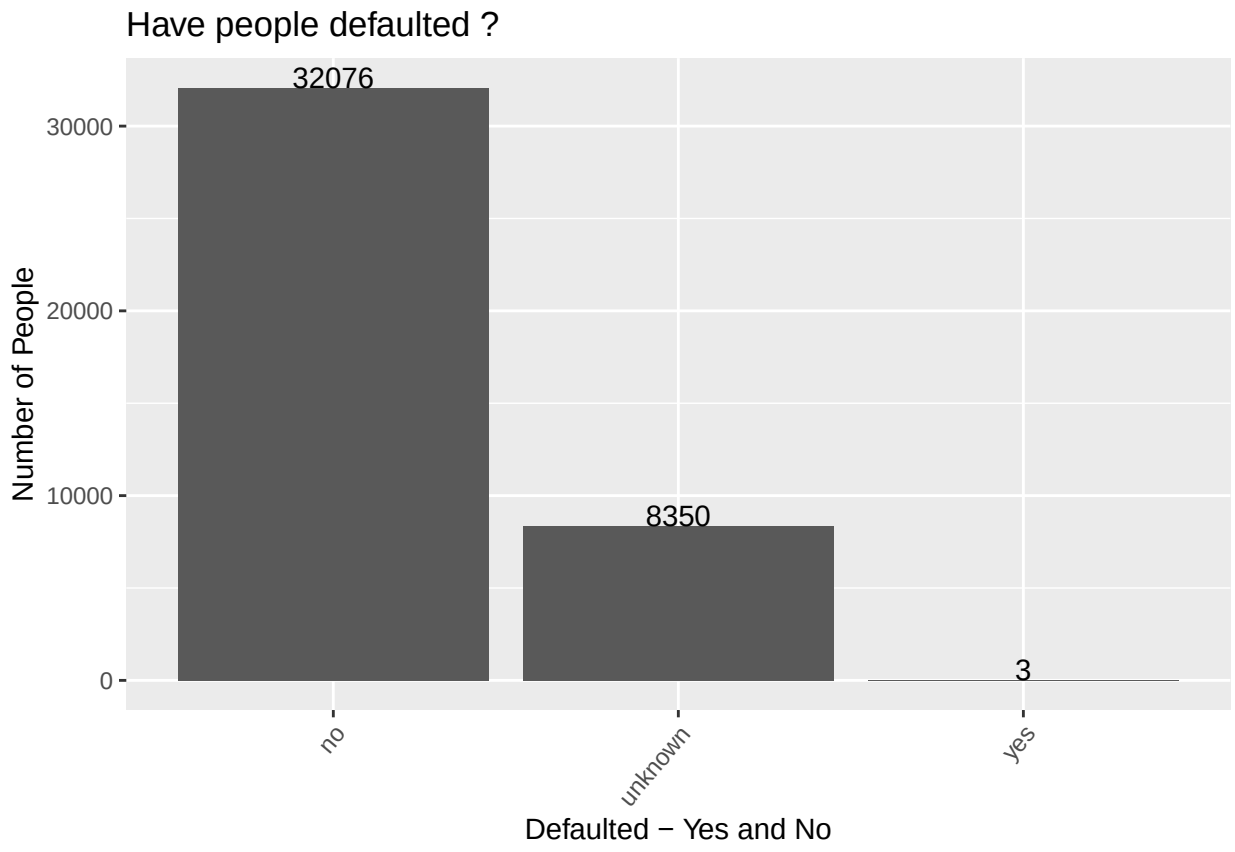
```
data = data %>% filter(education!="unknown")
nrow(data%>%filter(education=="unknown"))/nrow(data)*100
```

```
## [1] 0
```

```
data %>% count(default)
```

```
## # A tibble: 3 x 2
##   default     n
##   <chr>   <int>
## 1 no      32076
## 2 unknown  8350
## 3 yes         3
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(default))) +
  geom_bar()+
  stat_count(aes(label=..count..),
             vjust=0,
             geom="text",
             position="identity")+
  theme(axis.text.x=element_text(angle=50, hjust=1))+
  labs(title="Have people defaulted ?",
       x="Defaulted - Yes and No",
       y="Number of People")
```




```
nrow(data%>%filter(default=="unknown"))/nrow(data)*100
```

```
## [1] 20.65349
```

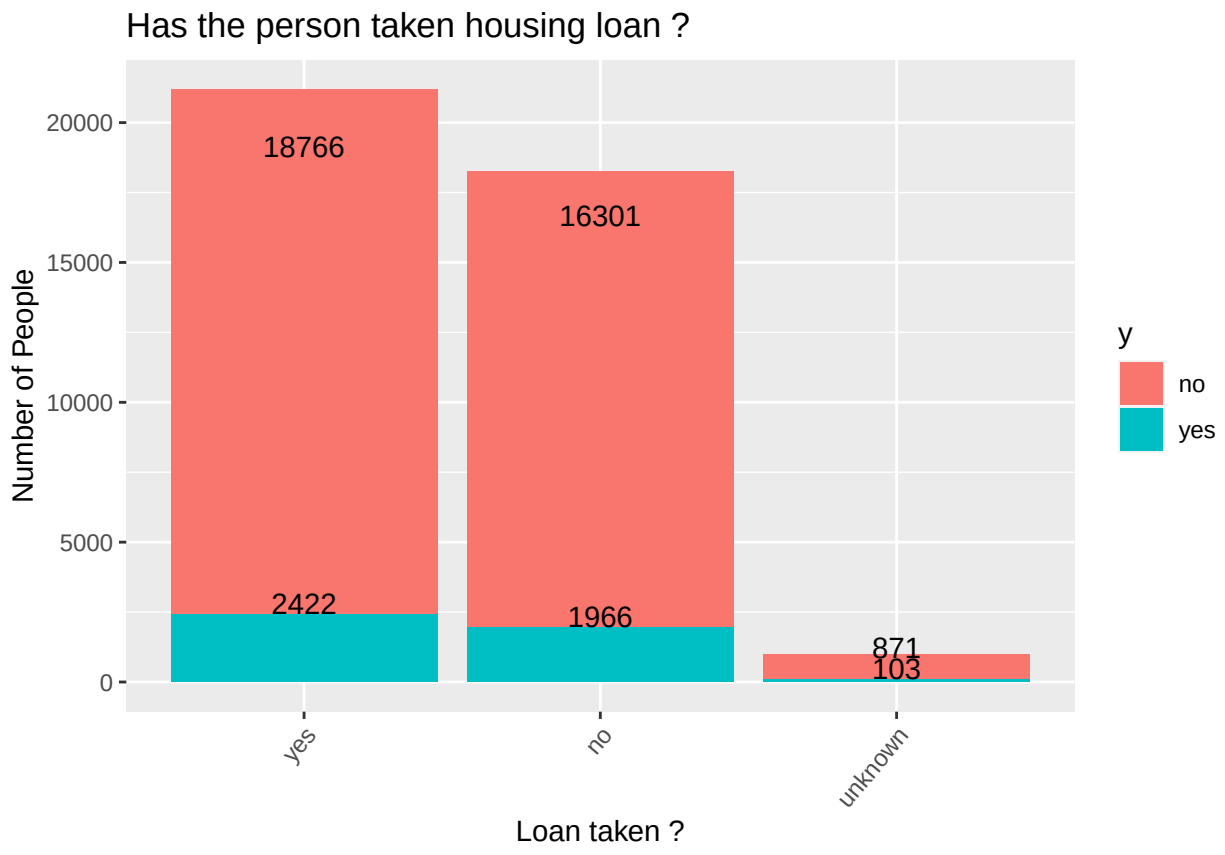
```
# data$default[data$default=="unknown"] <- "no"
nrow(data%>%filter(default=="unknown"))/nrow(data)*100
```

```
## [1] 20.65349
```

```
data %>% count(housing)
```

```
## # A tibble: 3 x 2
##   housing     n
##   <chr>   <int>
## 1 no      18267
## 2 unknown   974
## 3 yes     21188
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(housing), fill=y)) +
  geom_bar()+
  stat_count(aes(label=..count..),
             vjust=0,
             geom="text",
             position="identity")+
  theme(axis.text.x=element_text(angle=50, hjust=1))+
  labs(title="Has the person taken housing loan ?",
       x="Loan taken ?",
       y="Number of People")
```



```
nrow(data%>%filter(housing=="unknown"))/nrow(data)*100
```

```
## [1] 2.409162
```

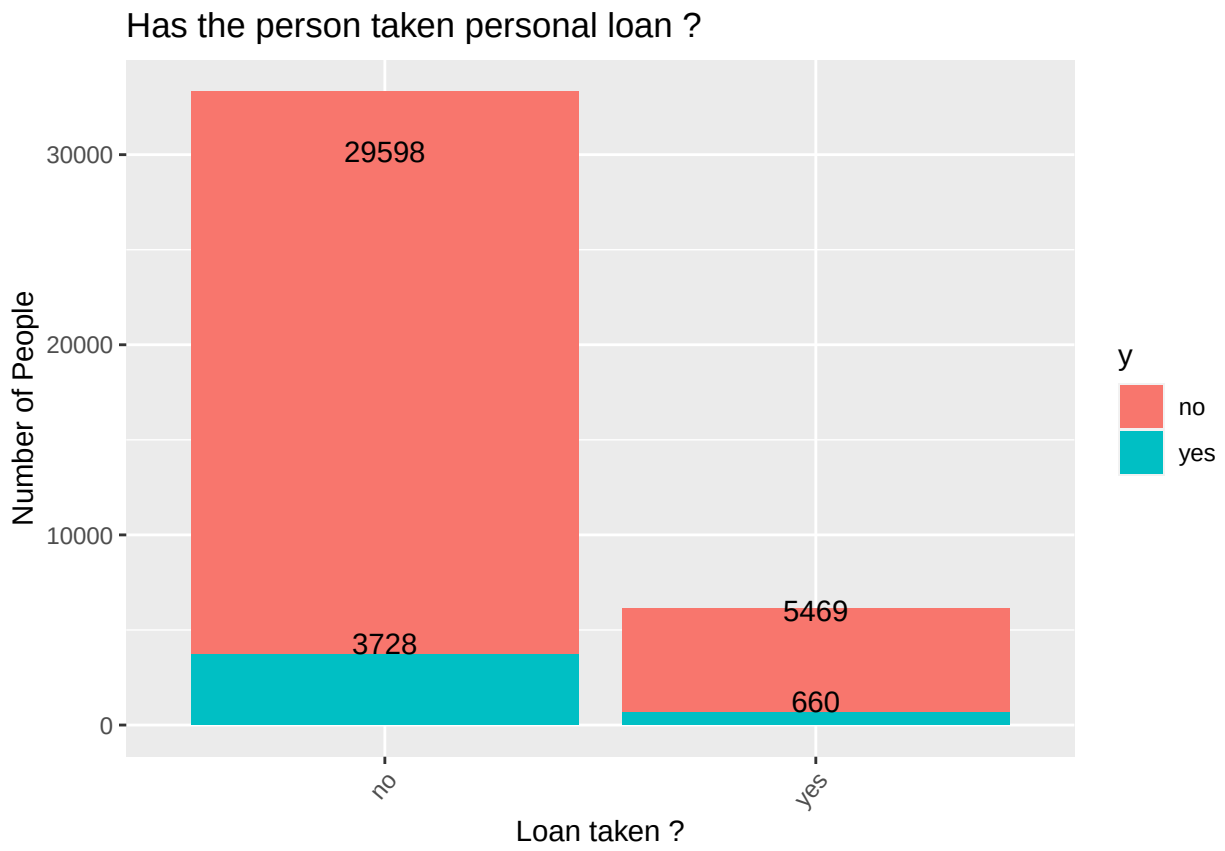
```
data = data %>% filter(housing!="unknown")
nrow(data%>%filter(housing=="unknown"))/nrow(data)*100
```

```
## [1] 0
```

```
data %>% count(loan)
```

```
## # A tibble: 2 x 2
##   loan      n
##   <chr> <int>
## 1 no    33326
## 2 yes    6129
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(loan), fill=y)) +
  geom_bar()+
  stat_count(aes(label=..count..),
             vjust=0,
             geom="text",
             position="identity")+
  theme(axis.text.x=element_text(angle=50, hjust=1))+
  labs(title="Has the person taken personal loan ?",
       x="Loan taken ?",
       y="Number of People")
```




```
nrow(data%>%filter(loan=="unknown"))/nrow(data)*100
```

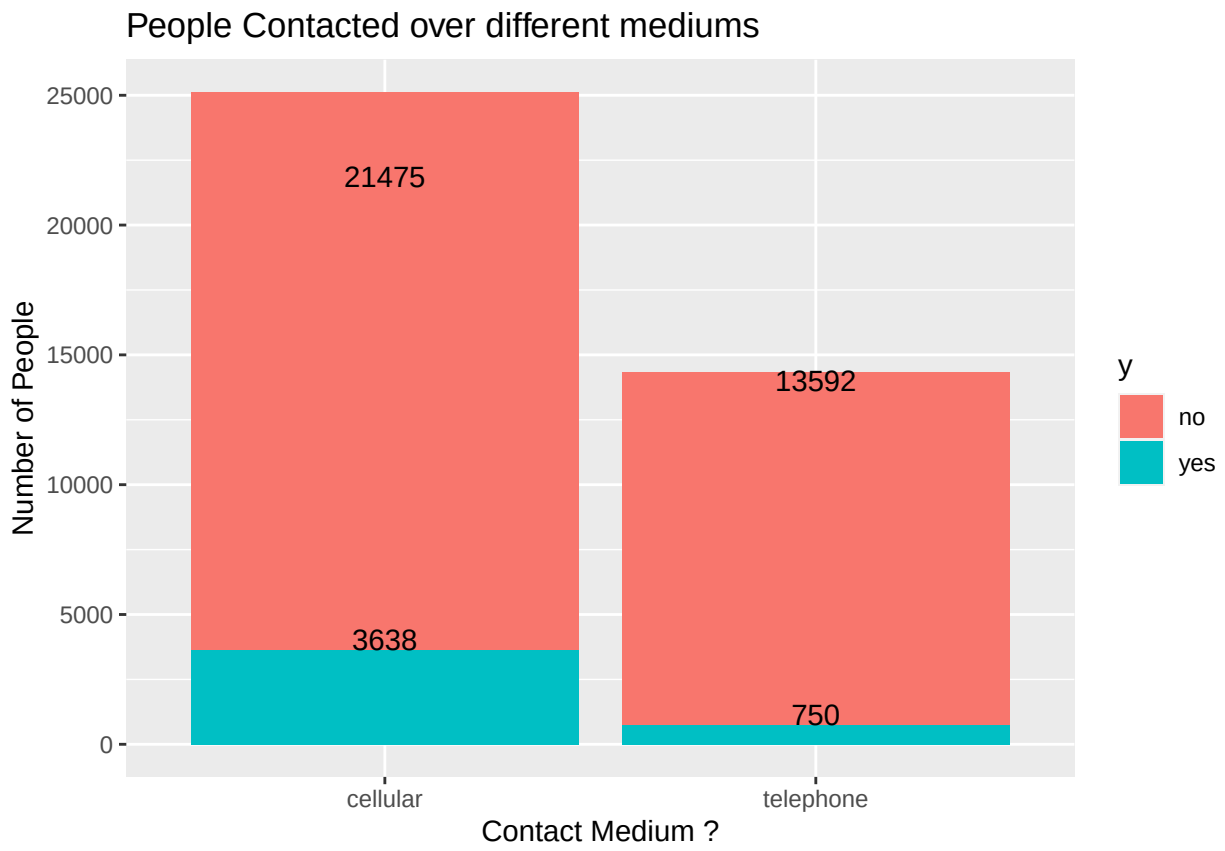
```
## [1] 0
```

Unknown values in the housing and personal loan columns correspond the same rows.

```
data %>% count(contact)
```

```
## # A tibble: 2 x 2
##   contact       n
##   <chr>     <int>
## 1 cellular  25113
## 2 telephone 14342
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(contact), fill=y)) +
  geom_bar()+
  stat_count(aes(label=..count..),
             vjust=0,
             geom="text",
             position="identity")+
  theme(axis.text.x=element_text( hjust=0.5))+
  labs(title="People Contacted over different mediums",
       x="Contact Medium ?",
       y="Number of People")
```

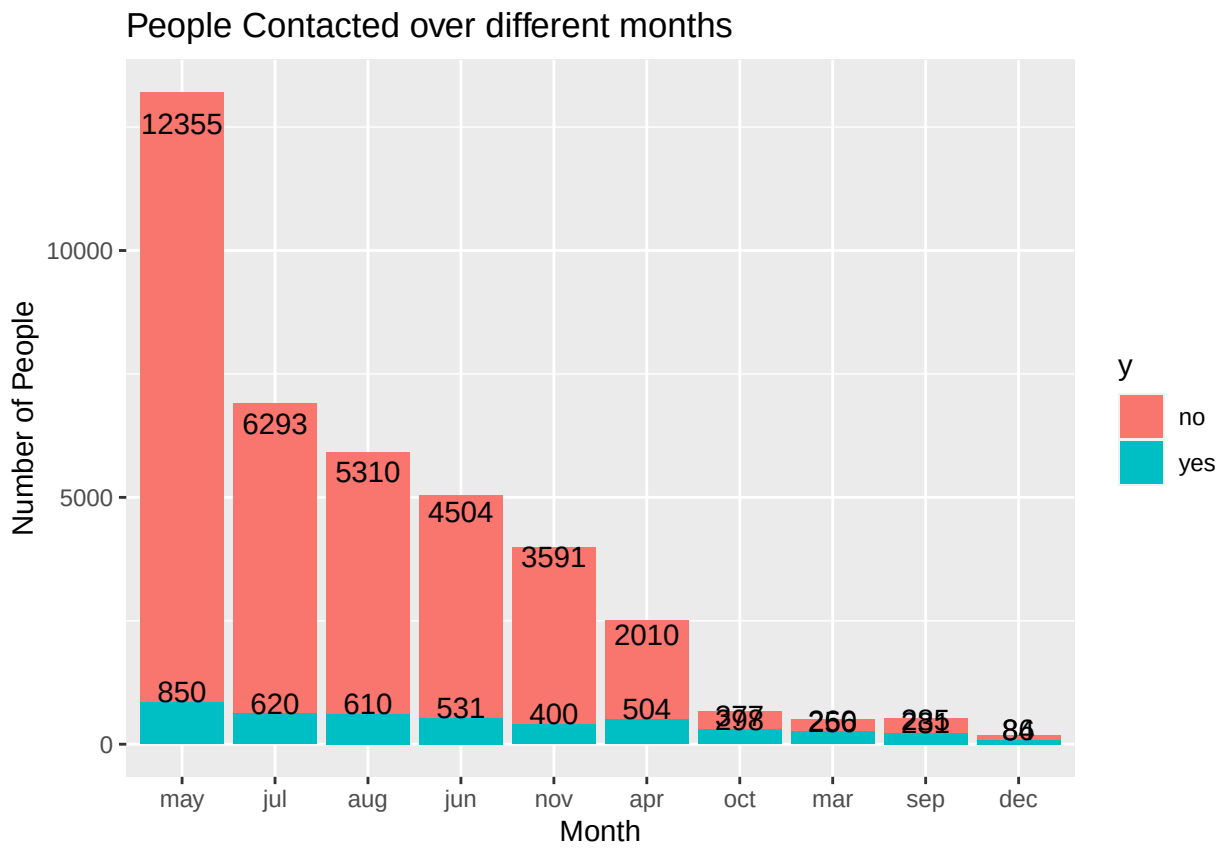



```
data %>% count(month)
```

```
## # A tibble: 10 x 2
##    month     n
##    <chr> <int>
##  1 apr    2514
##  2 aug    5920
##  3 dec     170
##  4 jul    6913
##  5 jun    5035
##  6 mar     516
##  7 may   13205
##  8 nov    3991
##  9 oct     675
## 10 sep     516
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(month), fill=y)) +
  geom_bar()+
  stat_count(aes(label=..count..),
             vjust=0,
             geom="text",
             position="identity")+
  theme(axis.text.x=element_text( hjust=0.5))+
  labs(title="People Contacted over different months",
```

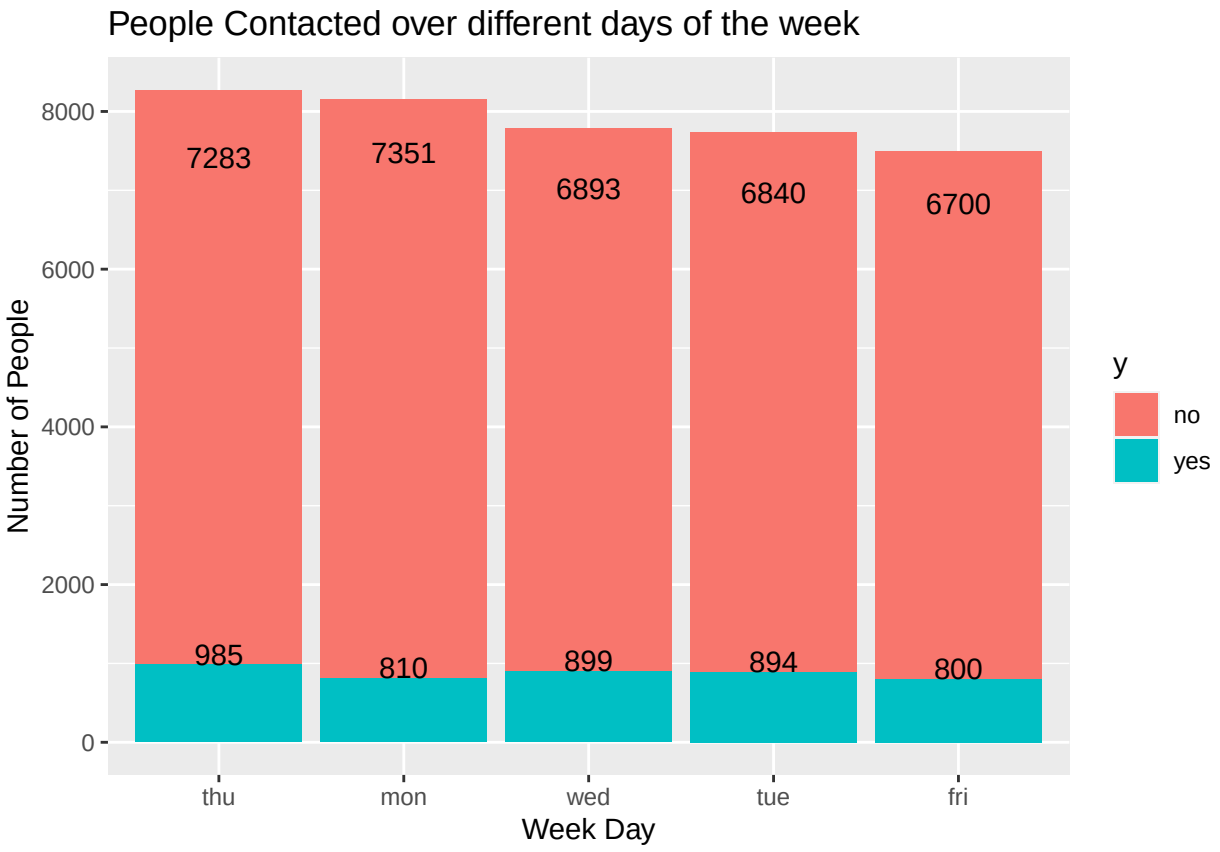
```
       x="Month",
       y="Number of People")
```




```
data %>% count(day_of_week)
```

```
## # A tibble: 5 x 2
##   day_of_week     n
##   <chr>       <int>
## 1 fri          7500
## 2 mon          8161
## 3 thu          8268
## 4 tue          7734
## 5 wed          7792
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(day_of_week), fill=y)) +
  geom_bar()+
  stat_count(aes(label=..count..),
             vjust=0,
             geom="text",
             position="identity")+
  theme(axis.text.x=element_text( hjust=0.5))+
  labs(title="People Contacted over different days of the week",
       x="Week Day",
       y="Number of People")
```




```
data %>% count(duration)
```

```
## # A tibble: 1,529 x 2
##    duration     n
##       <dbl> <int>
##  1        0     4
##  2        1     3
##  3        2     1
##  4        3     3
##  5        4    12
##  6        5    30
##  7        6    37
##  8        7    52
##  9        8    65
## 10        9    72
## # ... with 1,519 more rows
```

```
nrow(data%>%filter(duration==0))/nrow(data)*100
```

```
## [1] 0.01013813
```

First "yes" response was around 37 seconds call duration.

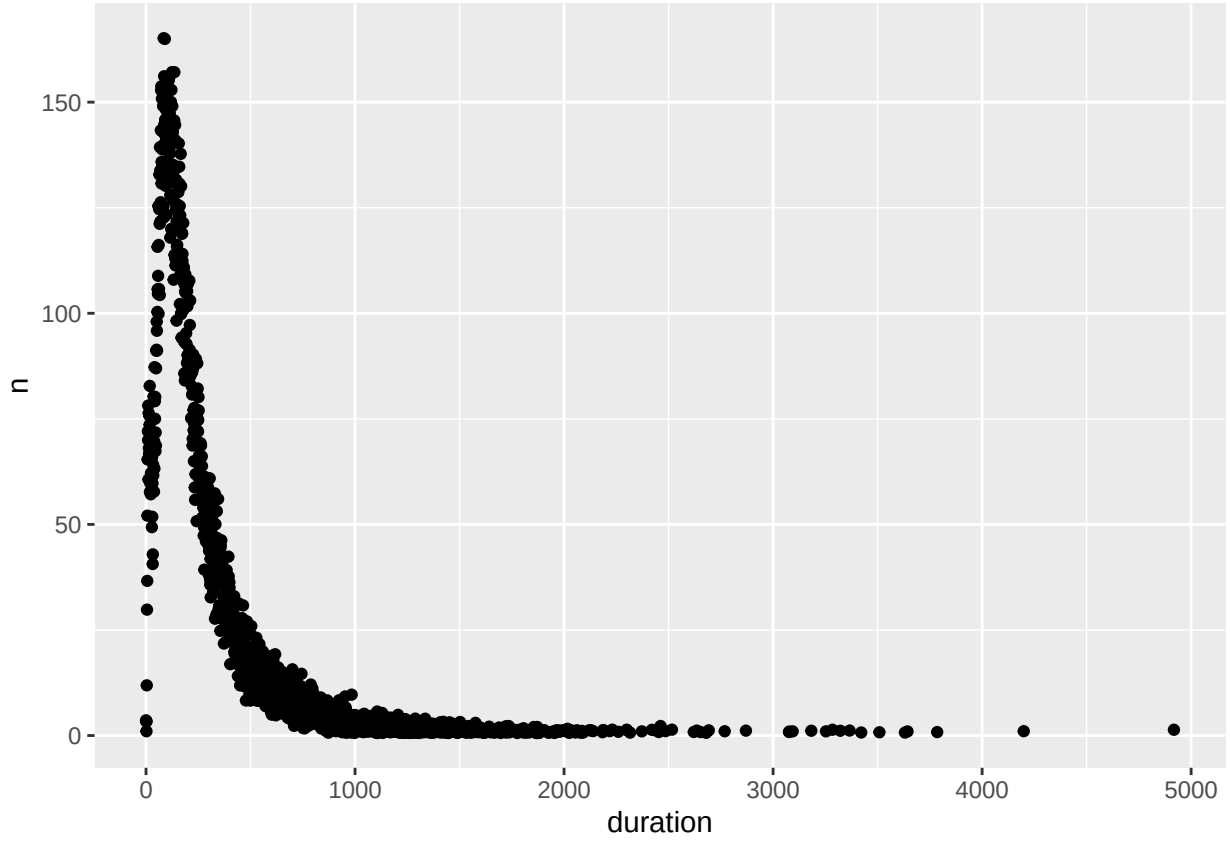
```
data %>% filter(y=="yes" & duration==37)
```

```
## # A tibble: 1 x 21
##     age job       marital education   default housing loan  contact month day_of_week
##   <dbl> <chr>     <chr>   <chr>       <chr>   <chr>   <chr> <chr>   <chr> <chr>
## 1    33 housemaid married high.school no      no      no    teleph~ oct   tue
## # ... with 11 more variables: duration <dbl>, campaign <dbl>, pdays <dbl>,
## #   previous <dbl>, poutcome <chr>, emp.var.rate <dbl>, cons.price.idx <dbl>,
## #   cons.conf.idx <dbl>, euribor3m <dbl>, nr.employed <dbl>, y <chr>
```

```
# temp = data %>% group_by(campaign,y ) %>% count()
# data = data %>% filter(duration>10)
# summary(data$duration)
```

Scatter plot - duration vs number of people:

```
data %>% group_by(duration) %>% count() %>%
ggplot(aes(x=duration,y=n))+
  geom_point(size=0.2)+
  geom_jitter()
```



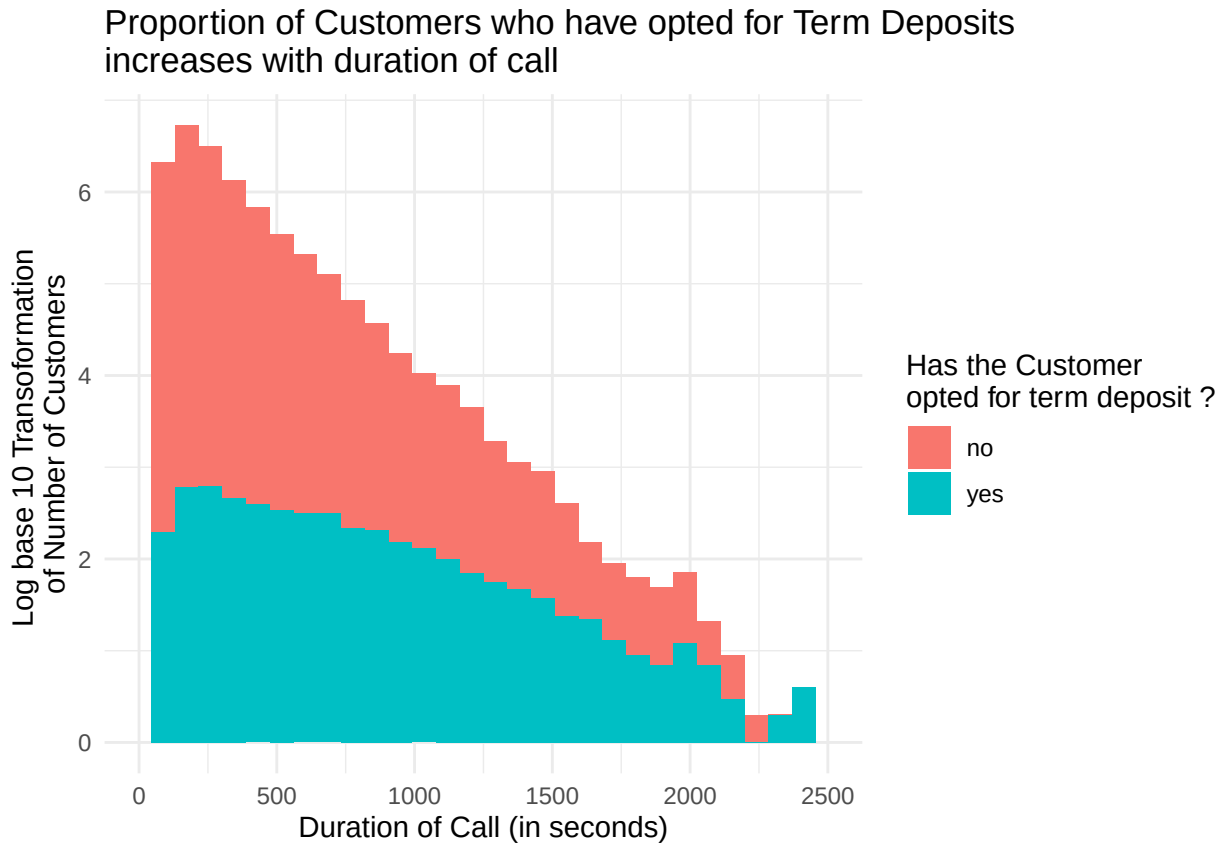

```
ggplot(data, aes(x=duration, fill=y))+
  geom_histogram(mapping = aes(y = after_stat(log10(count))))+
  labs(x="Duration of Call (in seconds)", y="Log base 10 Transoformation \nof Number of Customers",titl
```

```
  xlim(0,2500)+
  guides(fill=guide_legend(title="Has the Customer \nopted for term deposit ?"))+
  theme_minimal()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 23 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 4 rows containing missing values (geom_bar).
```




```
data %>% count(campaign)
```

```
## # A tibble: 41 x 2
##    campaign     n
##       <dbl> <int>
##  1        1 16882
##  2        2 10138
##  3        3  5115
##  4        4  2538
##  5        5  1543
##  6        6   941
##  7        7   605
##  8        8   381
```

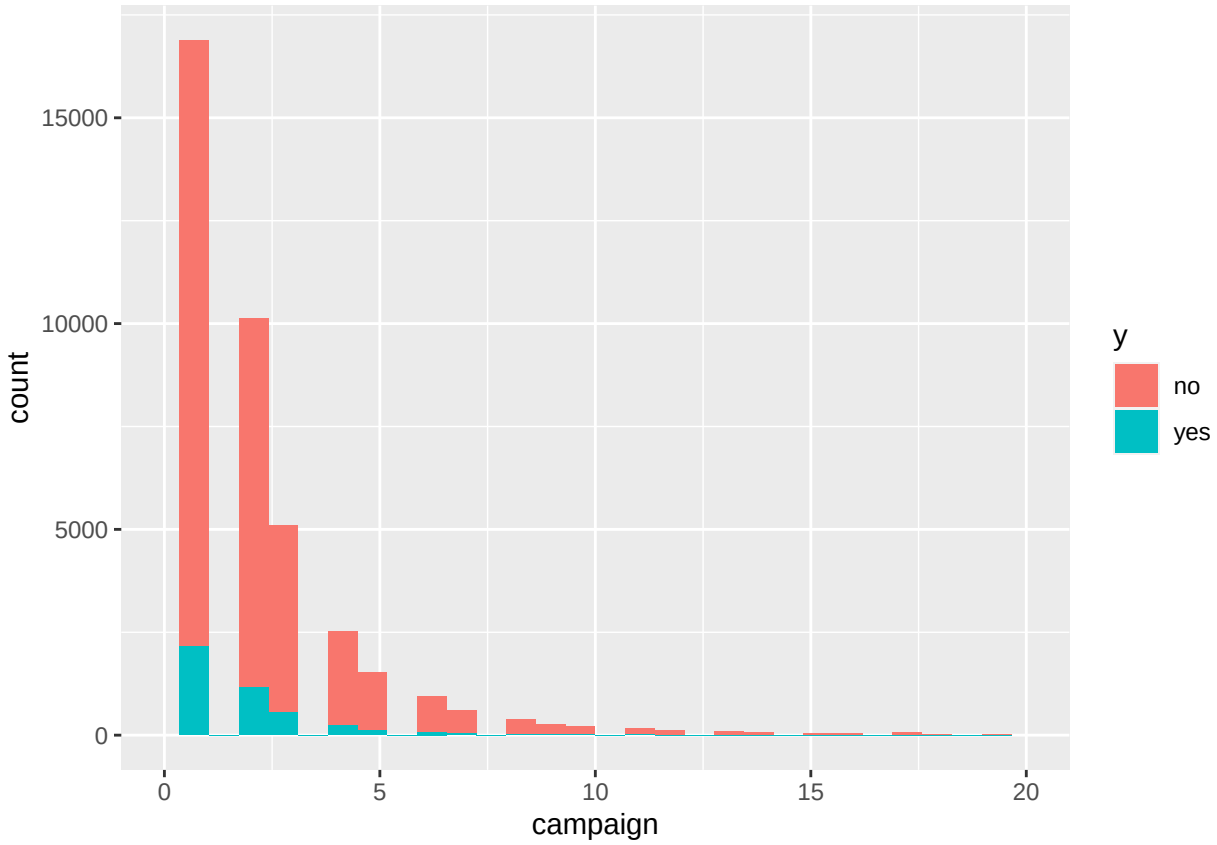
```
##  9        9   266
## 10       10   213
## # ... with 31 more rows
```

```
ggplot(data, aes(x=campaign, fill=y))+
  geom_histogram()+
  xlim(0,20)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 154 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 4 rows containing missing values (geom_bar).
```



Checking proportion of people converted over number of campaigns.

```
temp = data %>% group_by(campaign,y ) %>% count()
temp = temp %>% pivot_wider(names_from = y, values_from = n)
temp %>% mutate(p=yes/(yes+no))
```

```
## # A tibble: 41 x 4
## # Groups:   campaign [41]
##    campaign    no   yes      p
##       <dbl> <int> <int>  <dbl>
```

```
##  1        1 14725  2157 0.128
##  2        2  8985  1153 0.114
##  3        3  4566   549 0.107
##  4        4  2304   234 0.0922
##  5        5  1427   116 0.0752
##  6        6   868    73 0.0776
##  7        7   568    37 0.0612
##  8        8   365    16 0.0420
##  9        9   249    17 0.0639
## 10       10   203    10 0.0469
## # ... with 31 more rows
```

```
data %>% filter(campaign==56)
```

```
## # A tibble: 0 x 21
## # ... with 21 variables: age <dbl>, job <chr>, marital <chr>, education <chr>,
## #   default <chr>, housing <chr>, loan <chr>, contact <chr>, month <chr>,
## #   day_of_week <chr>, duration <dbl>, campaign <dbl>, pdays <dbl>,
## #   previous <dbl>, poutcome <chr>, emp.var.rate <dbl>, cons.price.idx <dbl>,
## #   cons.conf.idx <dbl>, euribor3m <dbl>, nr.employed <dbl>, y <chr>
```

The above person was contacted 56 times during previous campaigns but the call duration was 261 seconds and final response was NO.

```
data %>% count(pdays)
```

```
## # A tibble: 26 x 2
##    pdays     n
##    <dbl> <int>
##  1     0    15
##  2     1    25
##  3     2    57
##  4     3   407
##  5     4   109
##  6     5    45
##  7     6   391
##  8     7    55
##  9     8    15
## 10     9    60
## # ... with 16 more rows
```

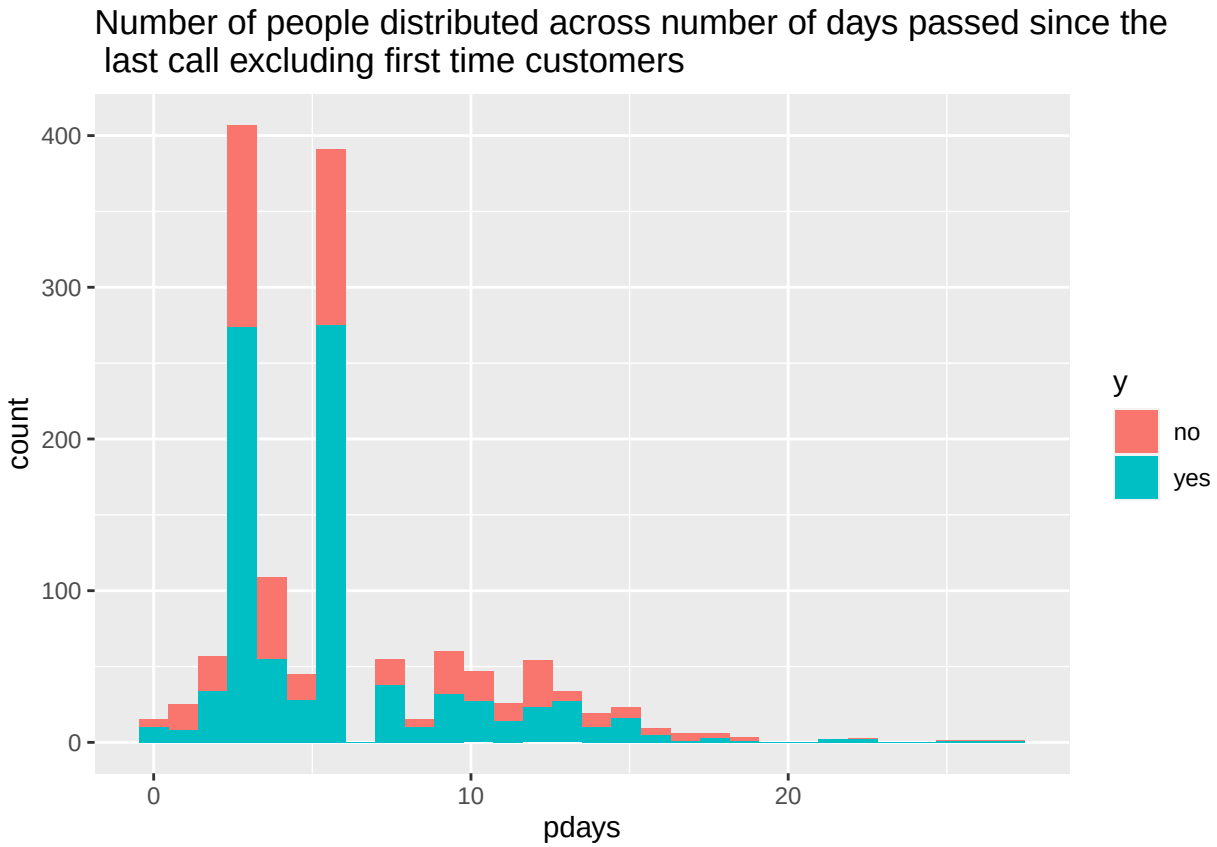
```
nrow(data%>%filter(pdays==999))/nrow(data)*100
```

```
## [1] 96.41617
```

Almost 96% of the customers haven't been contacted previously.

```
temp1 = data %>% filter(pdays!=999)
ggplot(temp1, aes(x=pdays, fill=y))+
  geom_histogram()+
  labs(title="Number of people distributed across number of days passed since the\n last call excluding
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
temp2 = temp1 %>% group_by(pdays,y ) %>% count()
temp2 = temp2 %>% pivot_wider(names_from = y, values_from = n)
temp2 %>% mutate(p=yes/(yes+no))
```

```
## # A tibble: 25 x 4
## # Groups:   pdays [25]
##    pdays    no   yes     p
##    <dbl> <int> <int> <dbl>
##  1     0     5    10 0.667
##  2     1    17     8 0.32
##  3     2    23    34 0.596
##  4     3   133   274 0.673
##  5     4    54    55 0.505
##  6     5    17    28 0.622
##  7     6   116   275 0.703
##  8     7    17    38 0.691
##  9     8     5    10 0.667
## 10     9    28    32 0.533
## # ... with 15 more rows
```

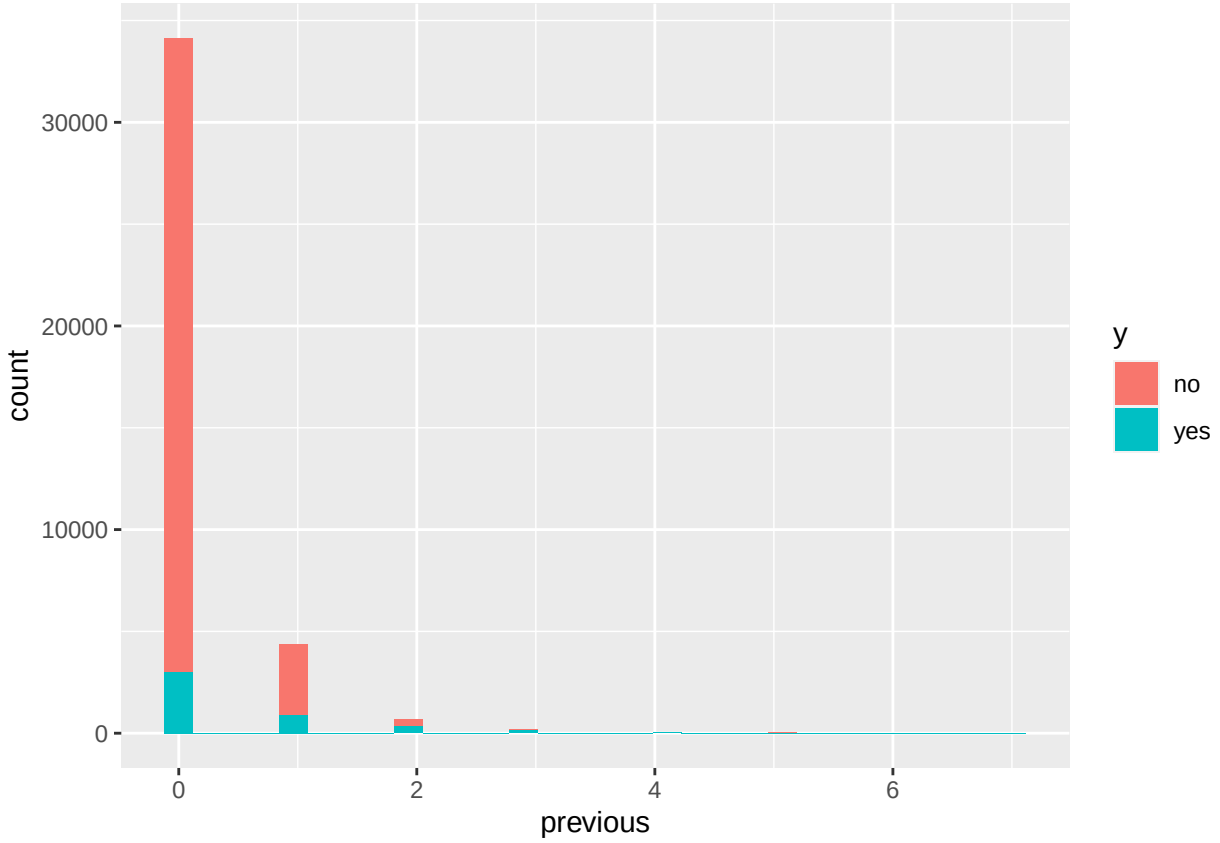
```
data %>% count(previous)
```

```
## # A tibble: 8 x 2
```

```
##   previous     n
##      <dbl> <int>
## 1        0 34122
## 2        1  4353
## 3        2   696
## 4        3   202
## 5        4    59
## 6        5    17
## 7        6     5
## 8        7     1
```

```
ggplot(data, aes(x=previous, fill=y))+
  geom_histogram()
```

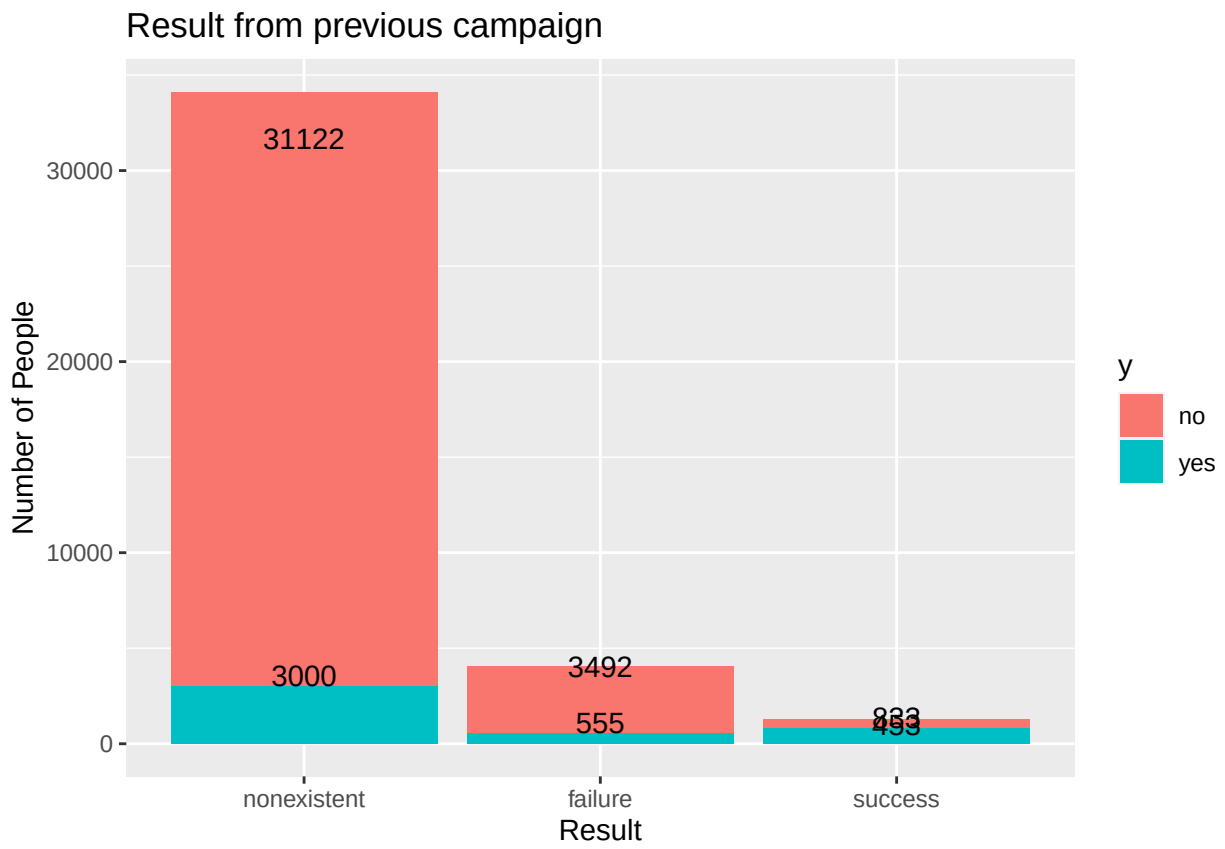
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
data %>% count(poutcome)
```

```
## # A tibble: 3 x 2
##   poutcome        n
##   <chr>       <int>
## 1 failure      4047
## 2 nonexistent 34122
## 3 success      1286
```

```
ggplot(data,
       aes(x = forcats::fct_infreq(poutcome), fill=y)) +
  geom_bar()+
  stat_count(aes(label=..count..),
             vjust=0,
             geom="text",
             position="identity")+
  theme(axis.text.x=element_text( hjust=0.5))+
  labs(title="Result from previous campaign",
       x="Result",
       y="Number of People")
```



Number of 'nonexistent' entries in 'poutcome' = 34122 = number of people who have been contacted previously 0 times

```
data %>% count(cons.price.idx)
```

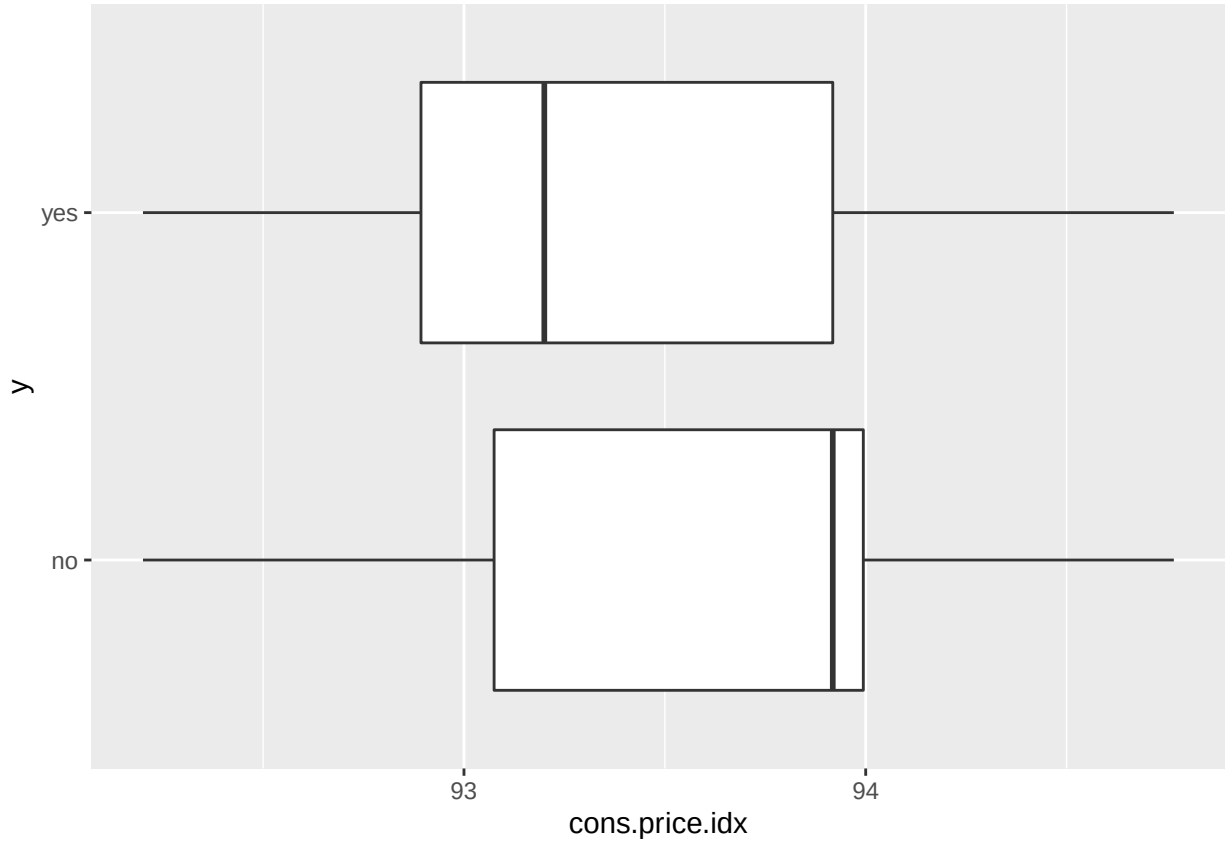
```
## # A tibble: 26 x 2
##    cons.price.idx     n
##             <dbl> <int>
##  1           92.2   711
##  2           92.4   244
##  3           92.4   420
##  4           92.5   166
##  5           92.6   336
```

```
##  6           92.7   160
##  7           92.8    10
##  8           92.8   272
##  9           92.9  5612
## 10           93.0   660
## # ... with 16 more rows
```

```
summary(data$cons.price.idx)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   92.20   93.08   93.75   93.57   93.99   94.77
```

```
ggplot(data, aes(x=cons.price.idx, y))+
  geom_boxplot()
```



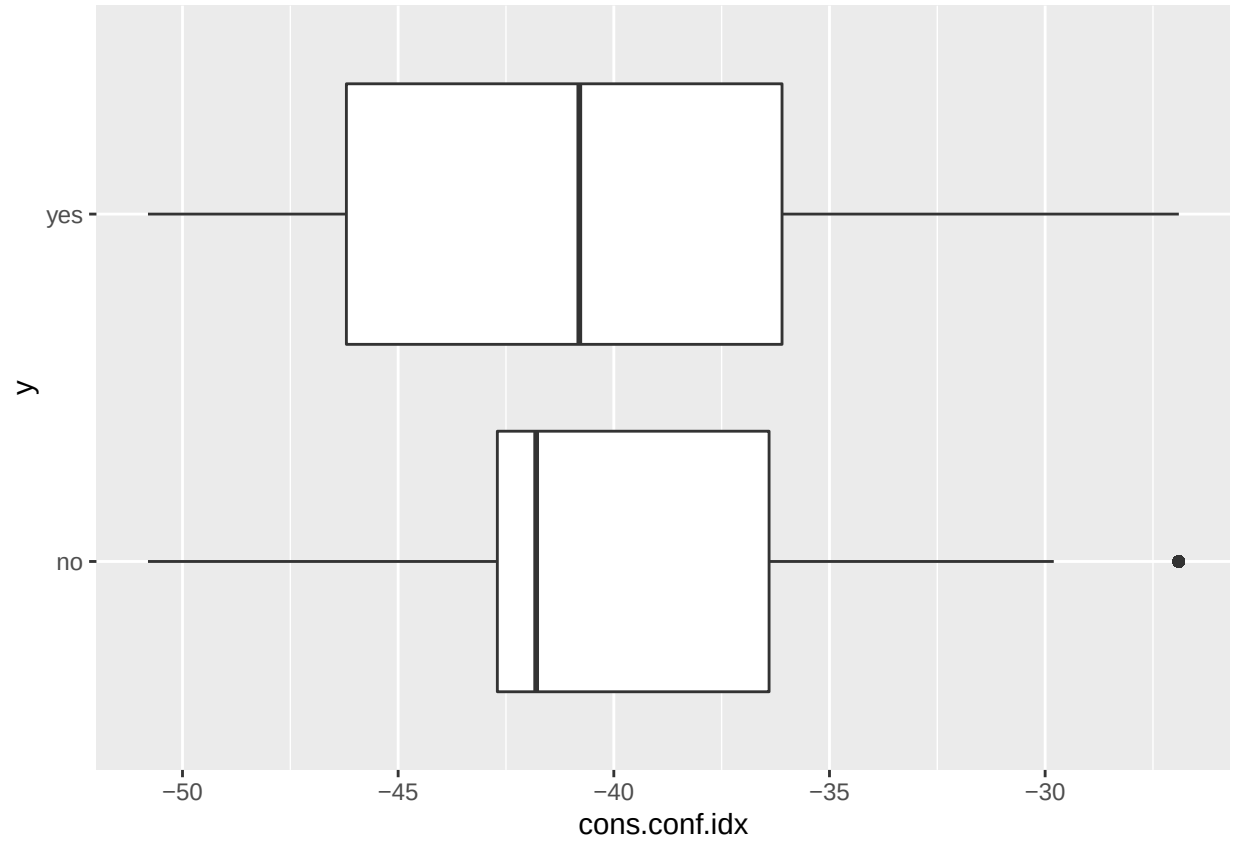
High price index -> people setting up the term deposit Low price index -> people not setting up the term deposit

```
data %>% count(cons.conf.idx)
```

```
## # A tibble: 26 x 2
##    cons.conf.idx     n
##            <dbl> <int>
##  1         -50.8   124
```

```
##  2         -50     272
##  3         -49.5   190
##  4         -47.1  2352
##  5         -46.2  5612
##  6         -45.9    10
##  7         -42.7  6457
##  8         -42    3531
##  9         -41.8  4154
## 10         -40.8   660
## # ... with 16 more rows
```

```
ggplot(data, aes(x=cons.conf.idx, y))+
  geom_boxplot()
```



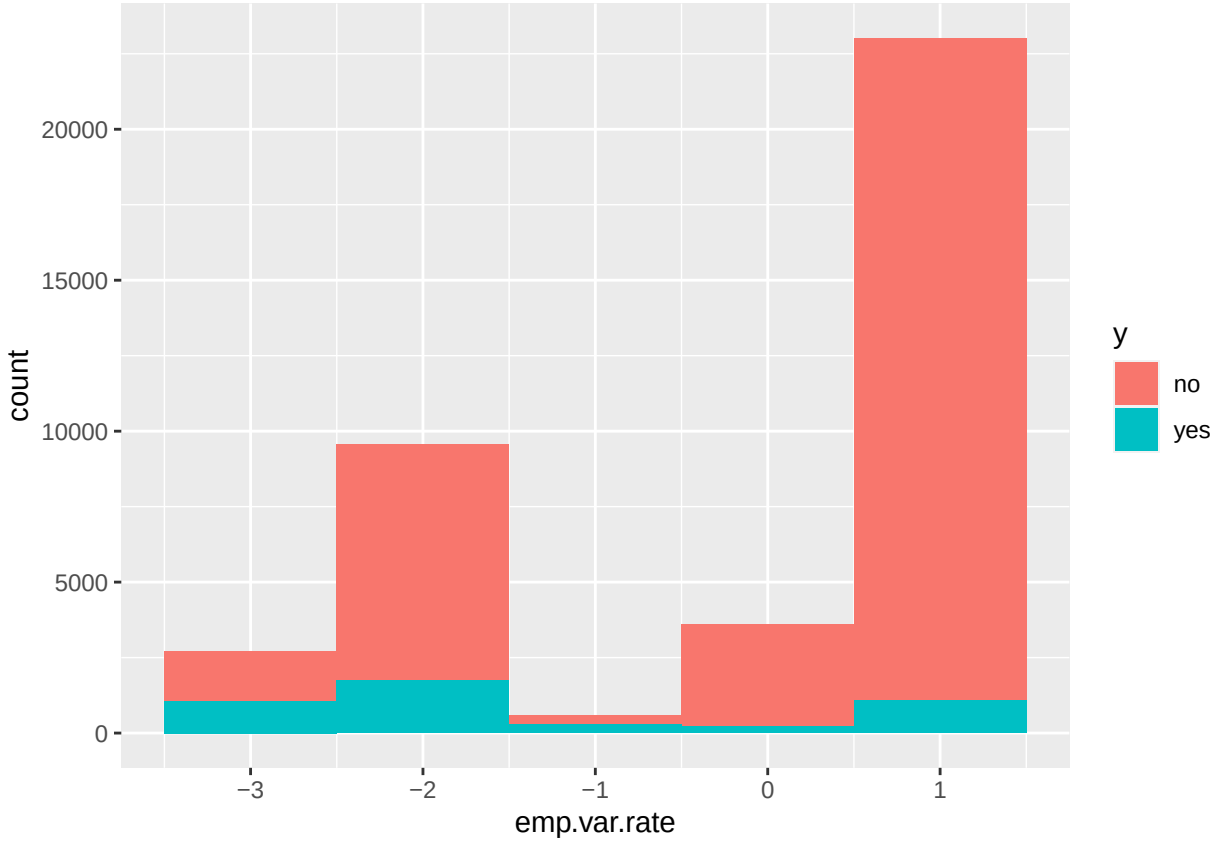
High confidence index -> people setting up the term deposit Low conf index -> people not setting up the term deposit

```
data %>% count(emp.var.rate)
```

```
## # A tibble: 10 x 2
##    emp.var.rate     n
##           <dbl> <int>
##  1         -3.4  1000
##  2         -3     160
##  3         -2.9  1537
```

```
##  4         -1.8  8831
##  5         -1.7   721
##  6         -1.1   586
##  7         -0.2    10
##  8         -0.1  3596
##  9          1.1  7404
## 10          1.4 15610
```

```
ggplot(data, aes(x=emp.var.rate, fill=y))+
  geom_histogram(binwidth = 1)
```




```
data %>% count(nr.employed)
```

```
## # A tibble: 11 x 2
##    nr.employed     n
##          <dbl> <int>
##  1       4964.   586
##  2       4992.   721
##  3       5009.   595
##  4       5018.  1000
##  5       5024.   160
##  6       5076.  1537
##  7       5099.  8236
##  8       5176.    10
##  9       5191   7404
```

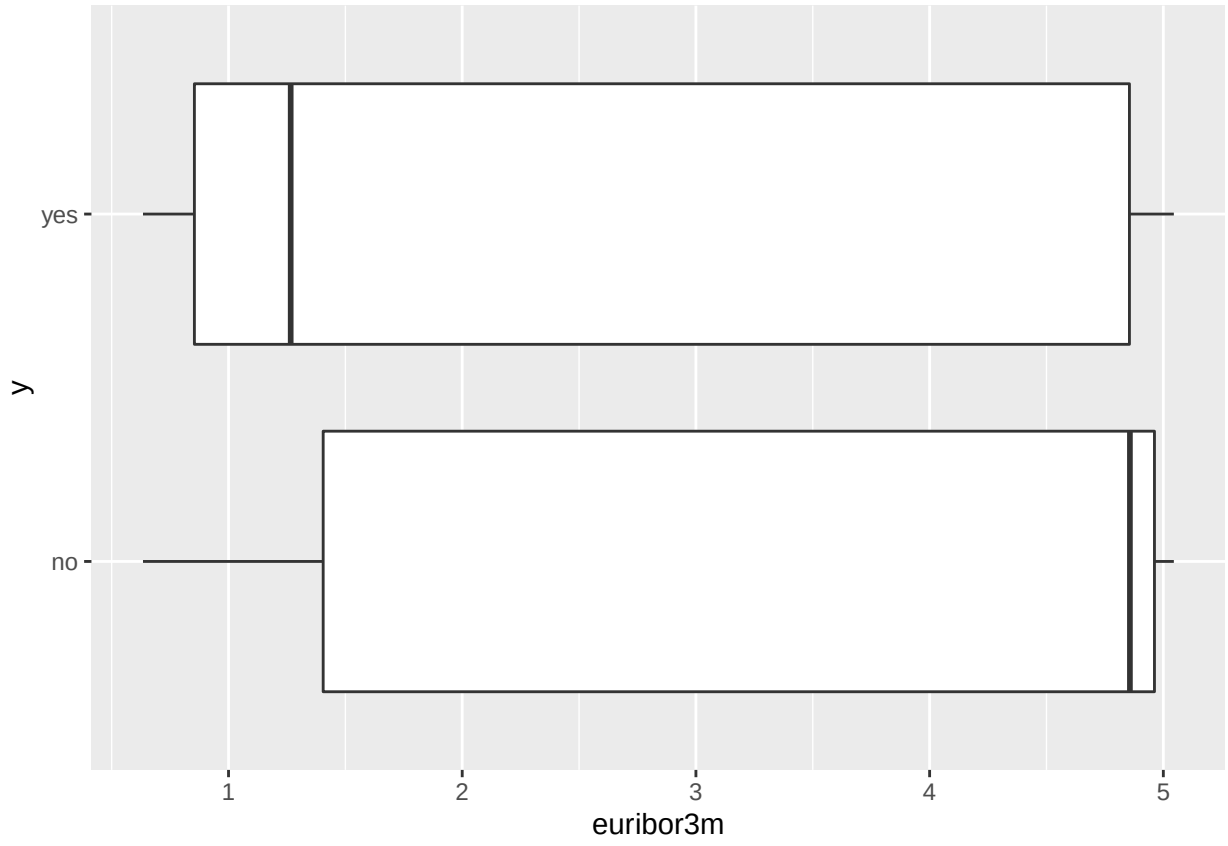
```
## 10       5196.  3596
## 11       5228. 15610
```

Employment Variation is low and its a quarterly indicator.

```
data %>% count(euribor3m)
```

```
## # A tibble: 316 x 2
##    euribor3m     n
##        <dbl> <int>
##  1     0.634     8
##  2     0.635    36
##  3     0.636    14
##  4     0.637     4
##  5     0.638     6
##  6     0.639    10
##  7     0.64      8
##  8     0.642    35
##  9     0.643    19
## 10     0.644    38
## # ... with 306 more rows
```

```
ggplot(data, aes(x=euribor3m, y))+
  geom_boxplot()
```



High Interest -> didn't set up the term deposit Low Interest -> set up the term deposit

Getting the filtered data frame:

```
filtered = data

#write.csv(filtered,"bank_filtered.csv", row.names = FALSE)
filtered
```

```
## # A tibble: 39,455 x 21
##      age job         marital education default housing loan  contact month day_of_week
##    <dbl> <chr>       <chr>   <chr>     <chr>   <chr>   <chr> <chr>   <chr> <chr>
##  1    56 housemaid   married basic.4y  no      no      no    teleph~ may   mon
##  2    57 services    married high.sch~ unknown no      no    teleph~ may   mon
##  3    37 services    married high.sch~ no      yes     no    teleph~ may   mon
##  4    40 admin       married basic.6y  no      no      no    teleph~ may   mon
##  5    56 services    married high.sch~ no      no      yes   teleph~ may   mon
##  6    45 services    married basic.9y  unknown no      no    teleph~ may   mon
##  7    59 admin       married professi~ no      no      no    teleph~ may   mon
##  8    41 blue-collar married basic.9y  unknown no      no    teleph~ may   mon
##  9    24 technician  single  professi~ no      yes     no    teleph~ may   mon
## 10    25 services    single  high.sch~ no      yes     no    teleph~ may   mon
## # ... with 39,445 more rows, and 11 more variables: duration <dbl>,
## #   campaign <dbl>, pdays <dbl>, previous <dbl>, poutcome <chr>,
## #   emp.var.rate <dbl>, cons.price.idx <dbl>, cons.conf.idx <dbl>,
## #   euribor3m <dbl>, nr.employed <dbl>, y <chr>
```

Heat Map:

```
data1 = select_if(data, is.numeric)

data$y[data$y=="yes"] <- 1
data$y[data$y=="no"] <- 0

data1$y = as.numeric(as.character(data$y))
```

```
library(reshape2)
```

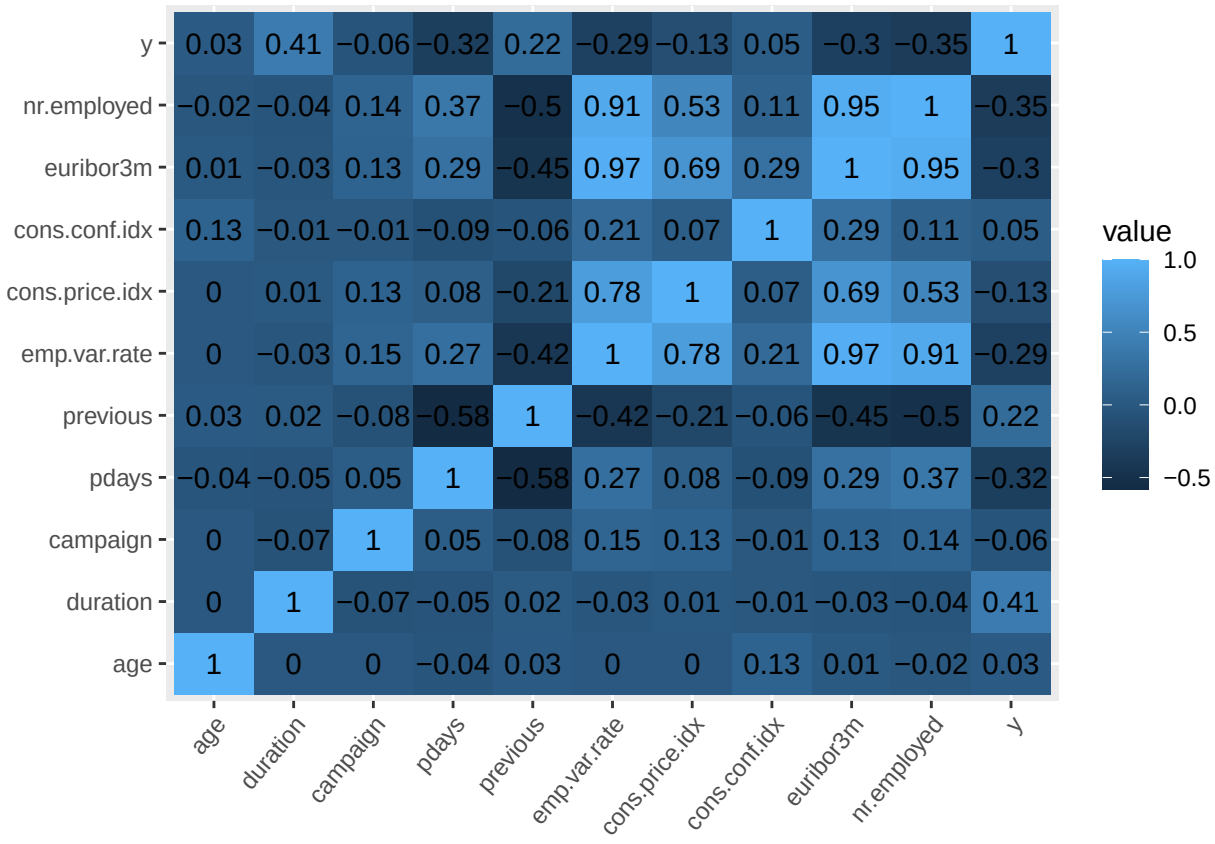
```
## Warning: package 'reshape2' was built under R version 4.1.2
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

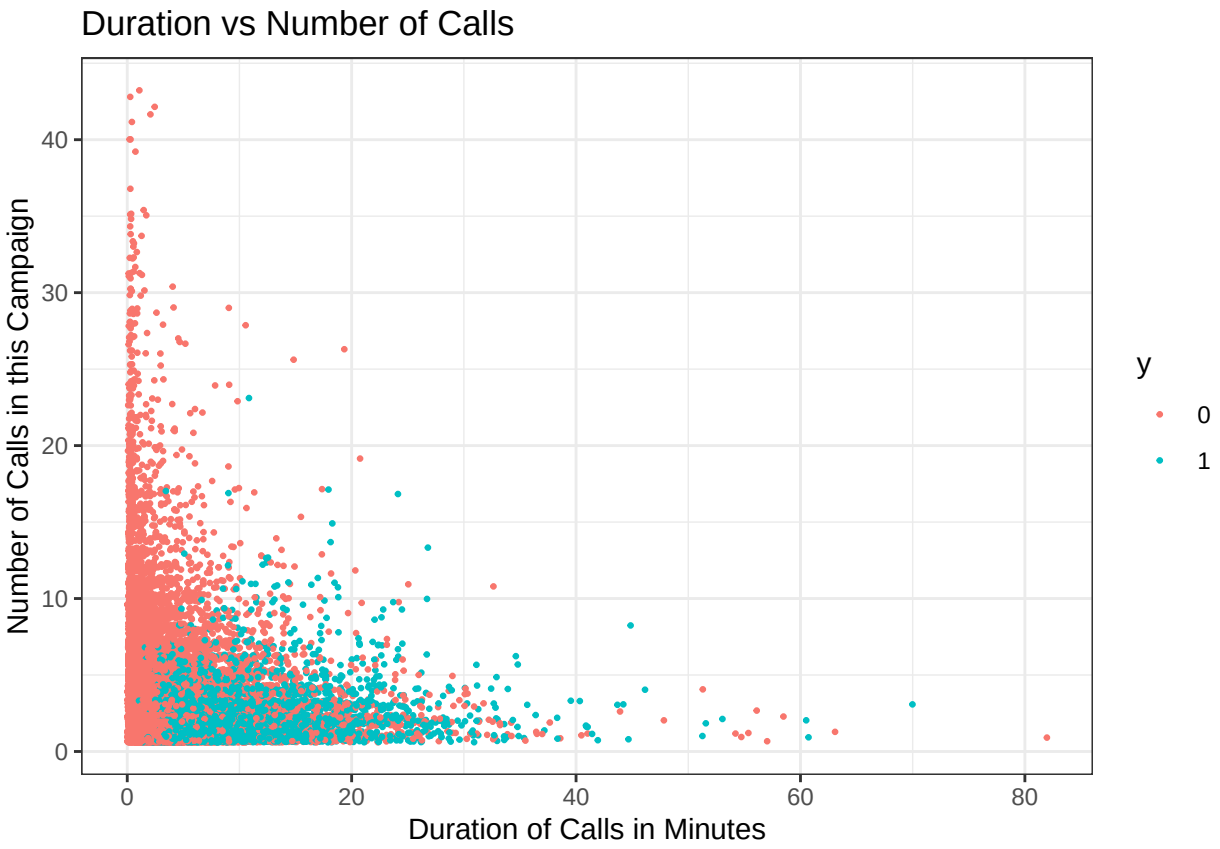
```
cor_matrix = round(cor(data1),2)
melted_matrix <- melt(cor_matrix)

ggplot(melted_matrix, aes(x=Var1, y=Var2, fill= value)) +
  geom_tile()+
```

```
geom_text(aes(Var2, Var1, label = value), color = "black", size = 4) +
theme(
  axis.text.x=element_text(angle=50, hjust=1),
  axis.title.x = element_blank(),
  axis.title.y = element_blank(),
  legend.direction = "vertical")
```



Additional

```
# Duration vs Campaign
data %>%
ggplot(aes(x=duration/60,y=campaign,color = y))+
  geom_jitter(size=0.5)+
  xlab("Duration of Calls in Minutes")+
  ylab("Number of Calls in this Campaign")+
  ggtitle("Duration vs Number of Calls")+
  theme_bw()
```



Correlation check

Categorical variables

```
library(dplyr)

col = c("job","marital","education","default","housing","loan","contact","month","day_of_week","poutcome

for (i in col){
  for (j in col){
    print(paste("P-value for ",i, "and",j,"is", round(chisq.test(get(i,data),get(j,data),simulate.p.valu
  }
}
```

```
## [1] "P-value for  job and job is 5e-04"
## [1] "P-value for  job and marital is 5e-04"
## [1] "P-value for  job and education is 5e-04"
## [1] "P-value for  job and default is 5e-04"
## [1] "P-value for  job and housing is 0.01799"
## [1] "P-value for  job and loan is 0.02299"
## [1] "P-value for  job and contact is 5e-04"
## [1] "P-value for  job and month is 5e-04"
## [1] "P-value for  job and day_of_week is 0.001"
```

```
## [1] "P-value for  job and poutcome is 5e-04"
## [1] "P-value for  job and y is 5e-04"
## [1] "P-value for  marital and job is 5e-04"
## [1] "P-value for  marital and marital is 5e-04"
## [1] "P-value for  marital and education is 5e-04"
## [1] "P-value for  marital and default is 5e-04"
## [1] "P-value for  marital and housing is 0.04248"
## [1] "P-value for  marital and loan is 0.51874"
## [1] "P-value for  marital and contact is 5e-04"
## [1] "P-value for  marital and month is 5e-04"
## [1] "P-value for  marital and day_of_week is 0.0035"
## [1] "P-value for  marital and poutcome is 5e-04"
## [1] "P-value for  marital and y is 5e-04"
## [1] "P-value for  education and job is 5e-04"
## [1] "P-value for  education and marital is 5e-04"
## [1] "P-value for  education and education is 5e-04"
## [1] "P-value for  education and default is 5e-04"
## [1] "P-value for  education and housing is 0.0035"
## [1] "P-value for  education and loan is 0.32484"
## [1] "P-value for  education and contact is 5e-04"
## [1] "P-value for  education and month is 5e-04"
## [1] "P-value for  education and day_of_week is 5e-04"
## [1] "P-value for  education and poutcome is 5e-04"
## [1] "P-value for  education and y is 5e-04"
## [1] "P-value for  default and job is 5e-04"
## [1] "P-value for  default and marital is 5e-04"
## [1] "P-value for  default and education is 5e-04"
## [1] "P-value for  default and default is 5e-04"
## [1] "P-value for  default and housing is 0.0025"
## [1] "P-value for  default and loan is 0.54923"
## [1] "P-value for  default and contact is 5e-04"
## [1] "P-value for  default and month is 5e-04"
## [1] "P-value for  default and day_of_week is 0.01699"
## [1] "P-value for  default and poutcome is 5e-04"
## [1] "P-value for  default and y is 5e-04"
## [1] "P-value for  housing and job is 0.01749"
## [1] "P-value for  housing and marital is 0.03798"
## [1] "P-value for  housing and education is 0.005"
## [1] "P-value for  housing and default is 0.0025"
## [1] "P-value for  housing and housing is 5e-04"
## [1] "P-value for  housing and loan is 5e-04"
## [1] "P-value for  housing and contact is 5e-04"
## [1] "P-value for  housing and month is 5e-04"
## [1] "P-value for  housing and day_of_week is 0.002"
## [1] "P-value for  housing and poutcome is 5e-04"
## [1] "P-value for  housing and y is 0.03348"
## [1] "P-value for  loan and job is 0.02149"
## [1] "P-value for  loan and marital is 0.51574"
## [1] "P-value for  loan and education is 0.34183"
## [1] "P-value for  loan and default is 0.57421"
## [1] "P-value for  loan and housing is 5e-04"
## [1] "P-value for  loan and loan is 5e-04"
## [1] "P-value for  loan and contact is 0.03198"
## [1] "P-value for  loan and month is 0.03448"
```

```
## [1] "P-value for  loan and day_of_week is 0.14543"
## [1] "P-value for  loan and poutcome is 0.87306"
## [1] "P-value for  loan and y is 0.33833"
## [1] "P-value for  contact and job is 5e-04"
## [1] "P-value for  contact and marital is 5e-04"
## [1] "P-value for  contact and education is 5e-04"
## [1] "P-value for  contact and default is 5e-04"
## [1] "P-value for  contact and housing is 5e-04"
## [1] "P-value for  contact and loan is 0.03448"
## [1] "P-value for  contact and contact is 5e-04"
## [1] "P-value for  contact and month is 5e-04"
## [1] "P-value for  contact and day_of_week is 5e-04"
## [1] "P-value for  contact and poutcome is 5e-04"
## [1] "P-value for  contact and y is 5e-04"
## [1] "P-value for  month and job is 5e-04"
## [1] "P-value for  month and marital is 5e-04"
## [1] "P-value for  month and education is 5e-04"
## [1] "P-value for  month and default is 5e-04"
## [1] "P-value for  month and housing is 5e-04"
## [1] "P-value for  month and loan is 0.03148"
## [1] "P-value for  month and contact is 5e-04"
## [1] "P-value for  month and month is 5e-04"
## [1] "P-value for  month and day_of_week is 5e-04"
## [1] "P-value for  month and poutcome is 5e-04"
## [1] "P-value for  month and y is 5e-04"
## [1] "P-value for  day_of_week and job is 0.004"
## [1] "P-value for  day_of_week and marital is 0.0015"
## [1] "P-value for  day_of_week and education is 5e-04"
## [1] "P-value for  day_of_week and default is 0.02249"
## [1] "P-value for  day_of_week and housing is 5e-04"
## [1] "P-value for  day_of_week and loan is 0.14743"
## [1] "P-value for  day_of_week and contact is 5e-04"
## [1] "P-value for  day_of_week and month is 5e-04"
## [1] "P-value for  day_of_week and day_of_week is 5e-04"
## [1] "P-value for  day_of_week and poutcome is 0.001"
## [1] "P-value for  day_of_week and y is 5e-04"
## [1] "P-value for  poutcome and job is 5e-04"
## [1] "P-value for  poutcome and marital is 5e-04"
## [1] "P-value for  poutcome and education is 5e-04"
## [1] "P-value for  poutcome and default is 5e-04"
## [1] "P-value for  poutcome and housing is 5e-04"
## [1] "P-value for  poutcome and loan is 0.87506"
## [1] "P-value for  poutcome and contact is 5e-04"
## [1] "P-value for  poutcome and month is 5e-04"
## [1] "P-value for  poutcome and day_of_week is 0.0015"
## [1] "P-value for  poutcome and poutcome is 5e-04"
## [1] "P-value for  poutcome and y is 5e-04"
## [1] "P-value for  y and job is 5e-04"
## [1] "P-value for  y and marital is 5e-04"
## [1] "P-value for  y and education is 5e-04"
## [1] "P-value for  y and default is 5e-04"
## [1] "P-value for  y and housing is 0.03648"
## [1] "P-value for  y and loan is 0.32334"
## [1] "P-value for  y and contact is 5e-04"
```

```
## [1] "P-value for  y and month is 5e-04"
## [1] "P-value for  y and day_of_week is 0.001"
## [1] "P-value for  y and poutcome is 5e-04"
## [1] "P-value for  y and y is 5e-04"
```

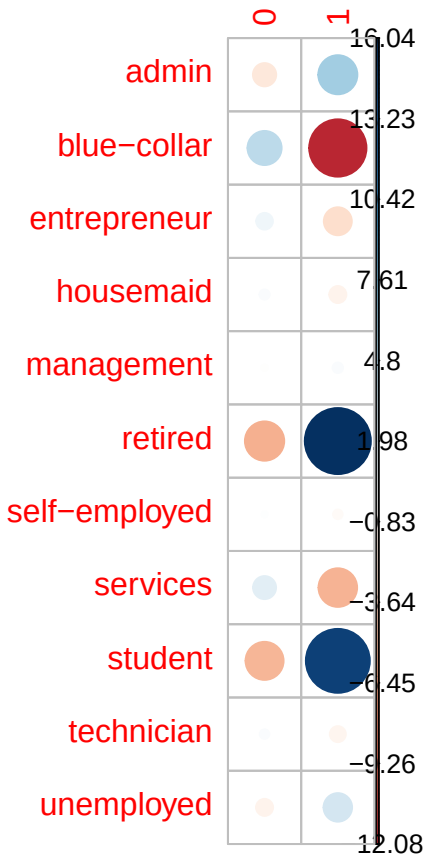
If we consider a threshold of 0.05 for p-value: loan can be removed.

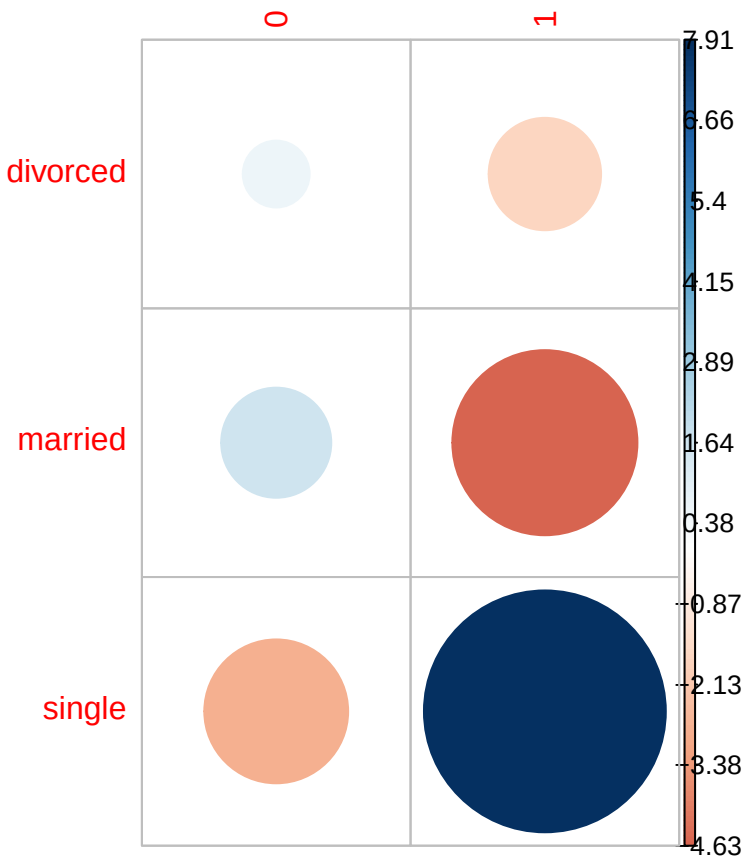
```
#install.packages("corrplot")
library(corrplot)
```

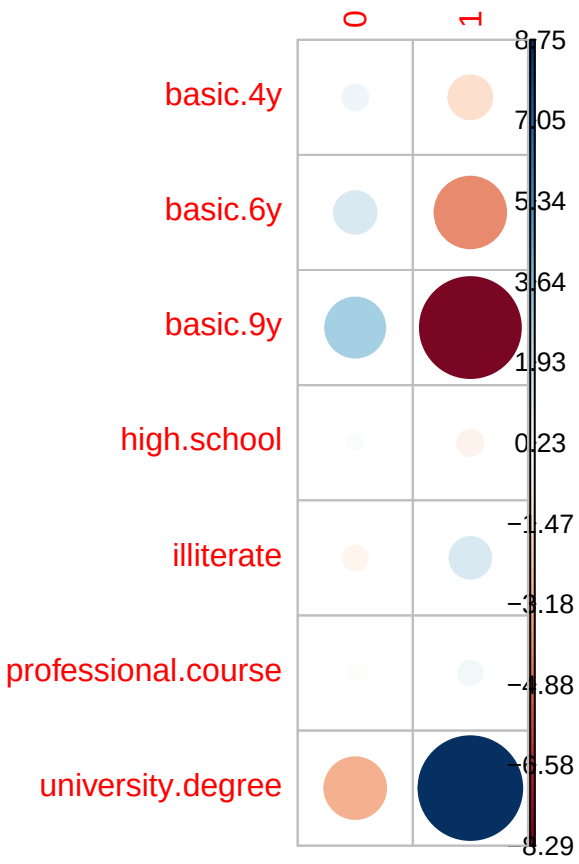
```
## Warning: package 'corrplot' was built under R version 4.1.2
```

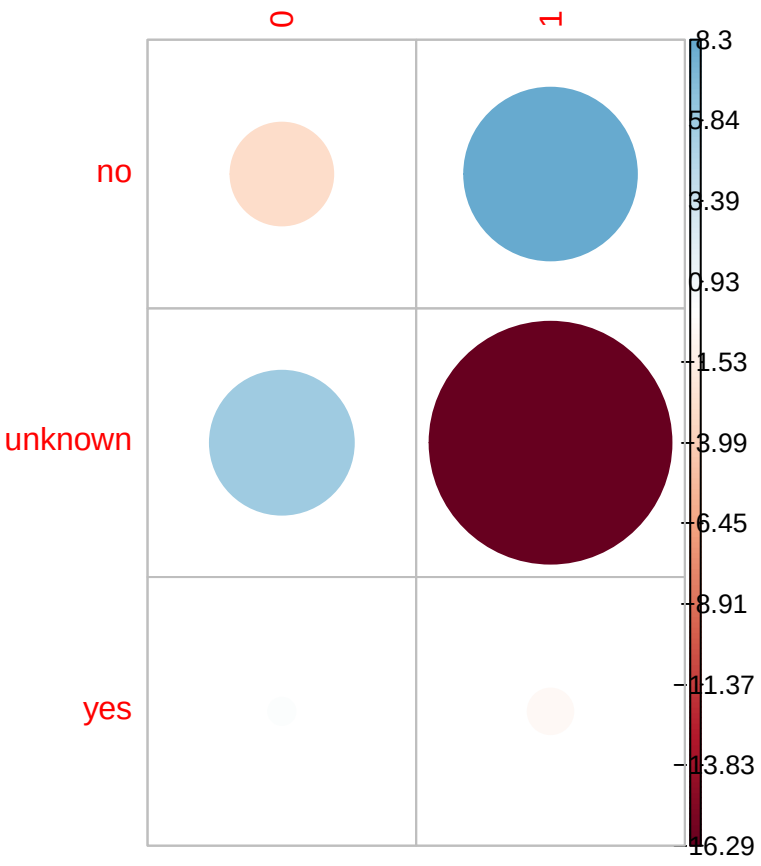
```
## corrplot 0.92 loaded
```

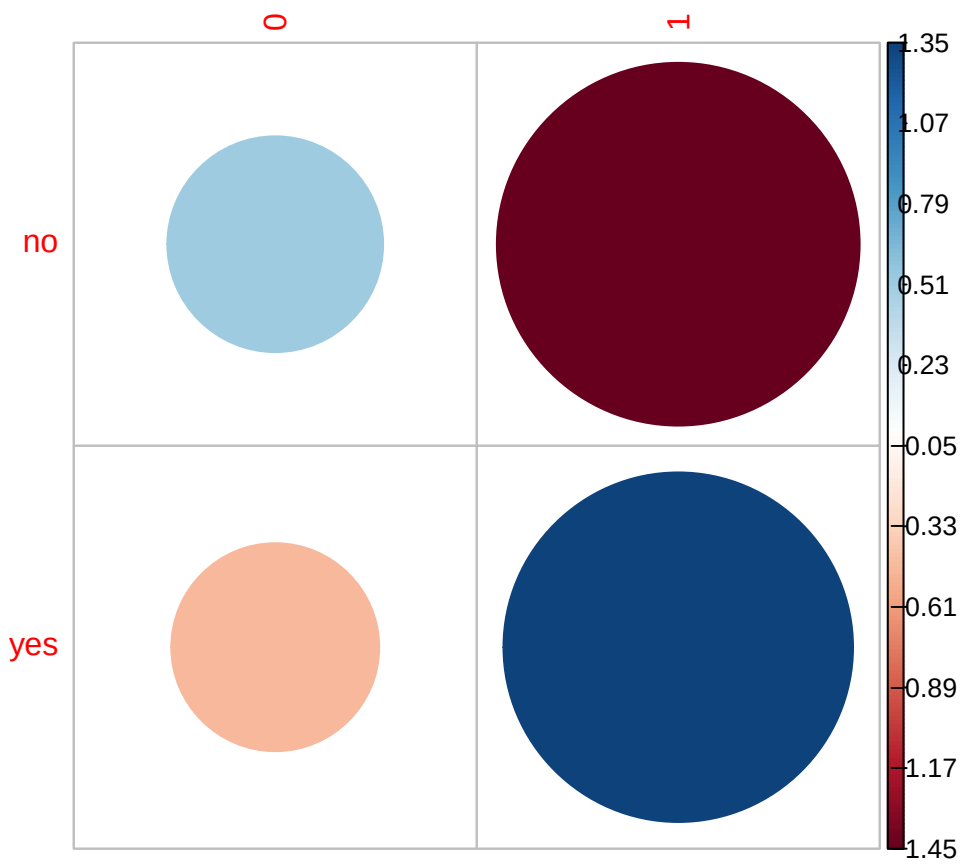
```
for (i in col){
  c <- chisq.test(get(i,data),data$y,simulate.p.value = TRUE)
  corrplot(c$residuals,is.cor=FALSE)
}
```

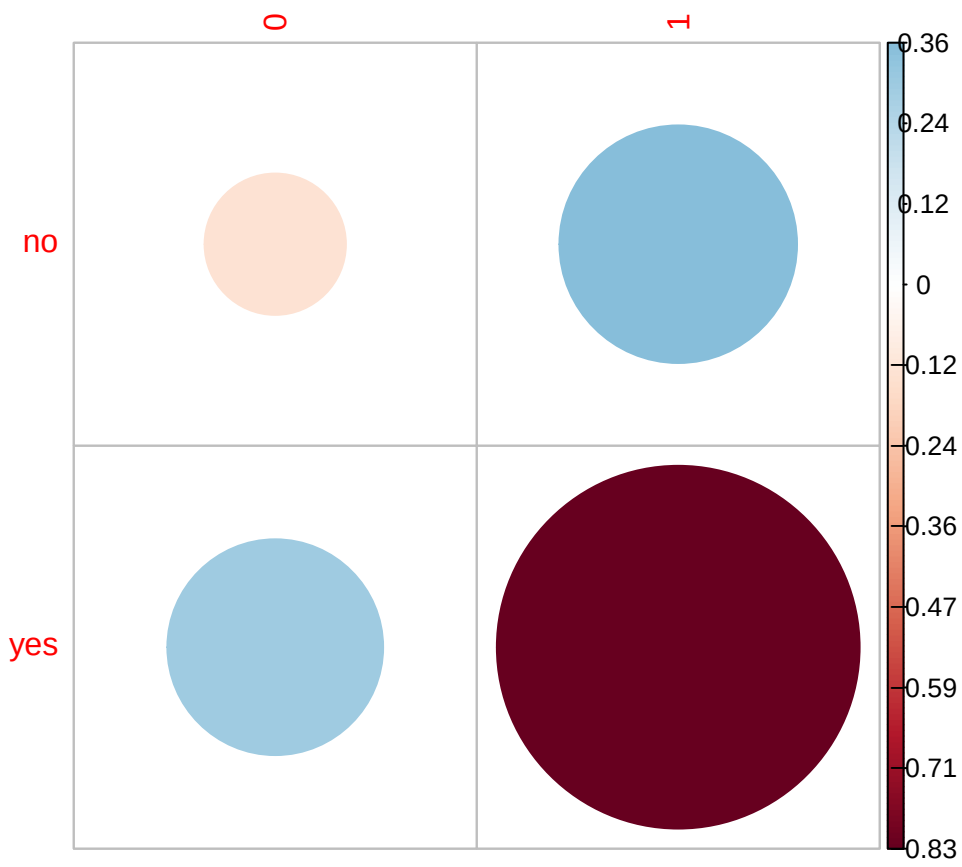


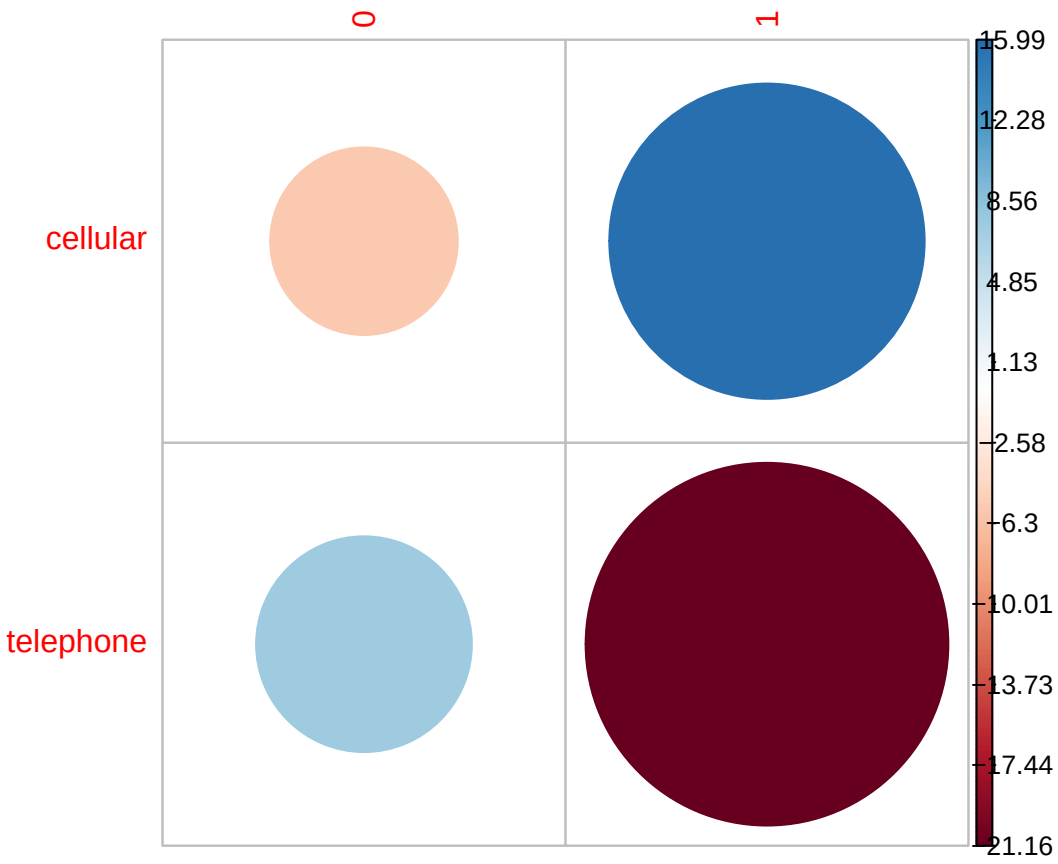

0
1
divorced
married
single
7.91
6.66
5.4
4.15
2.89
1.64
0.38
−0.87
−2.13
−3.38
−4.63

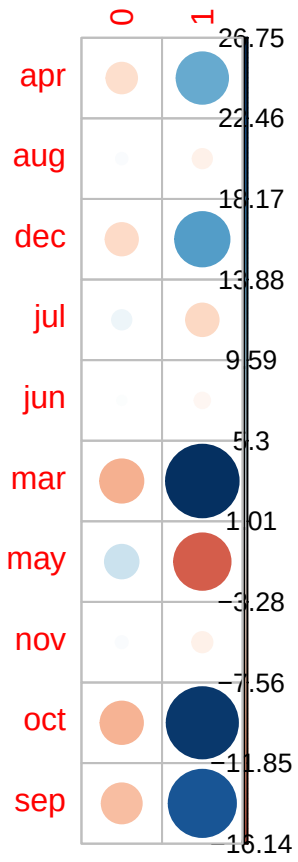

0
1
basic.4y
basic.6y
basic.9y
high.school
illiterate
professional.course
university.degree
8.75
7.05
5.34
3.64
1.93
0.23
−1.47
−3.18
−4.88
−6.58
−8.29

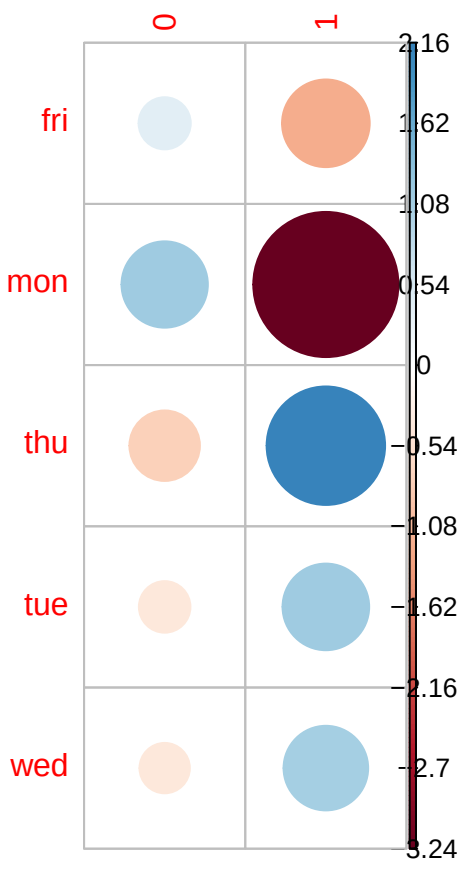

0
1
no
unknown
yes
8.3
5.84
3.39
0.93
−1.53
−3.99
−6.45
−8.91
−11.37
−13.83
16.29

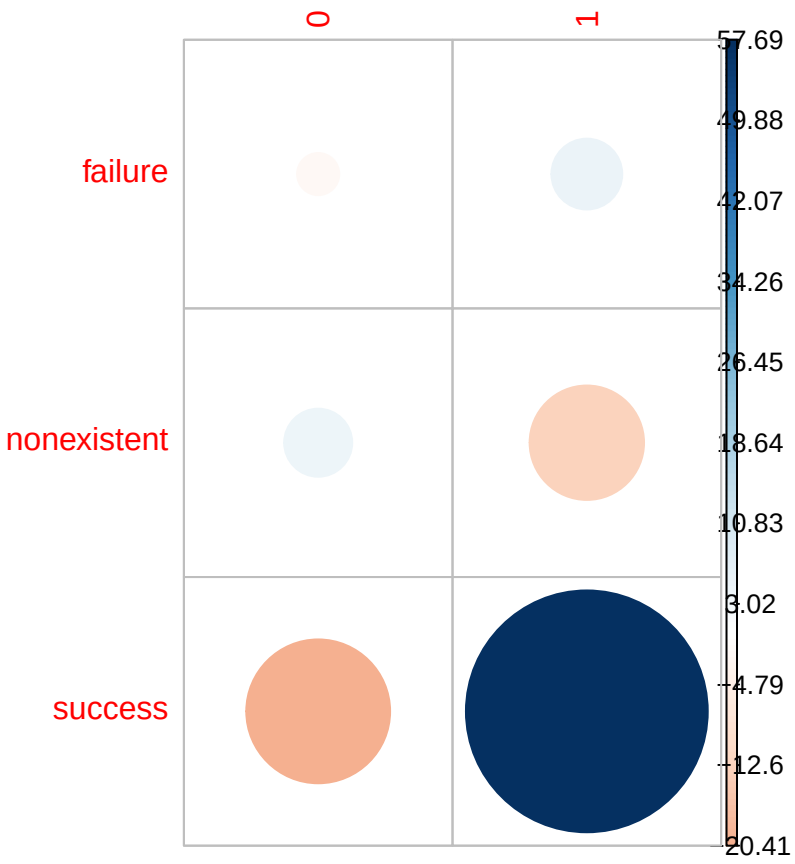

0
1
no
yes
1.35
1.07
0.79
0.51
0.23
0.05
0.33
0.61
0.89
1.17
1.45

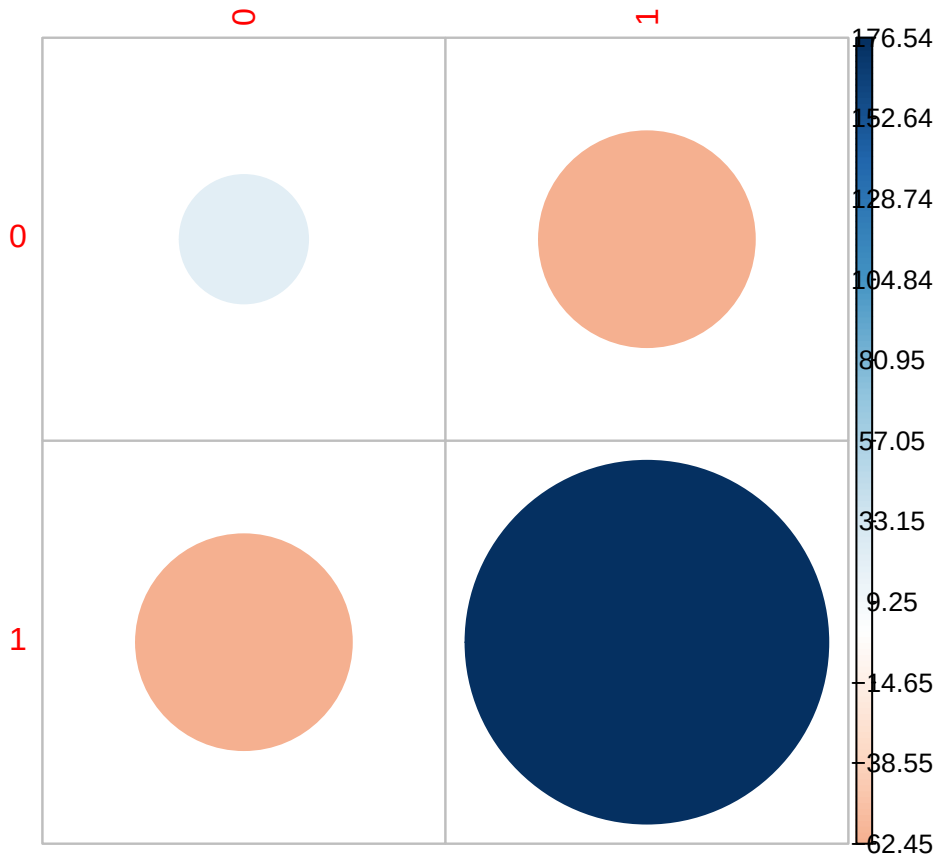

0
1
no
yes
0.36
0.24
0.12
0
0.12
0.24
0.36
0.47
0.59
0.71
0.83


0
1
cellular
telephone
15.99
12.28
8.56
4.85
1.13
−2.58
−6.3
−10.01
−13.73
−17.44
−21.16


0
1
apr
aug
dec
jul
jun
mar
may
nov
oct
sep
26.75
22.46
18.17
13.88
9.59
5.3
1.01
−3.28
−7.56
−11.85
−16.14


0
1
fri
mon
thu
tue
wed
2.16
1.62
1.08
0.54
0
−0.54
−1.08
−1.62
−2.16
−2.7
−3.24

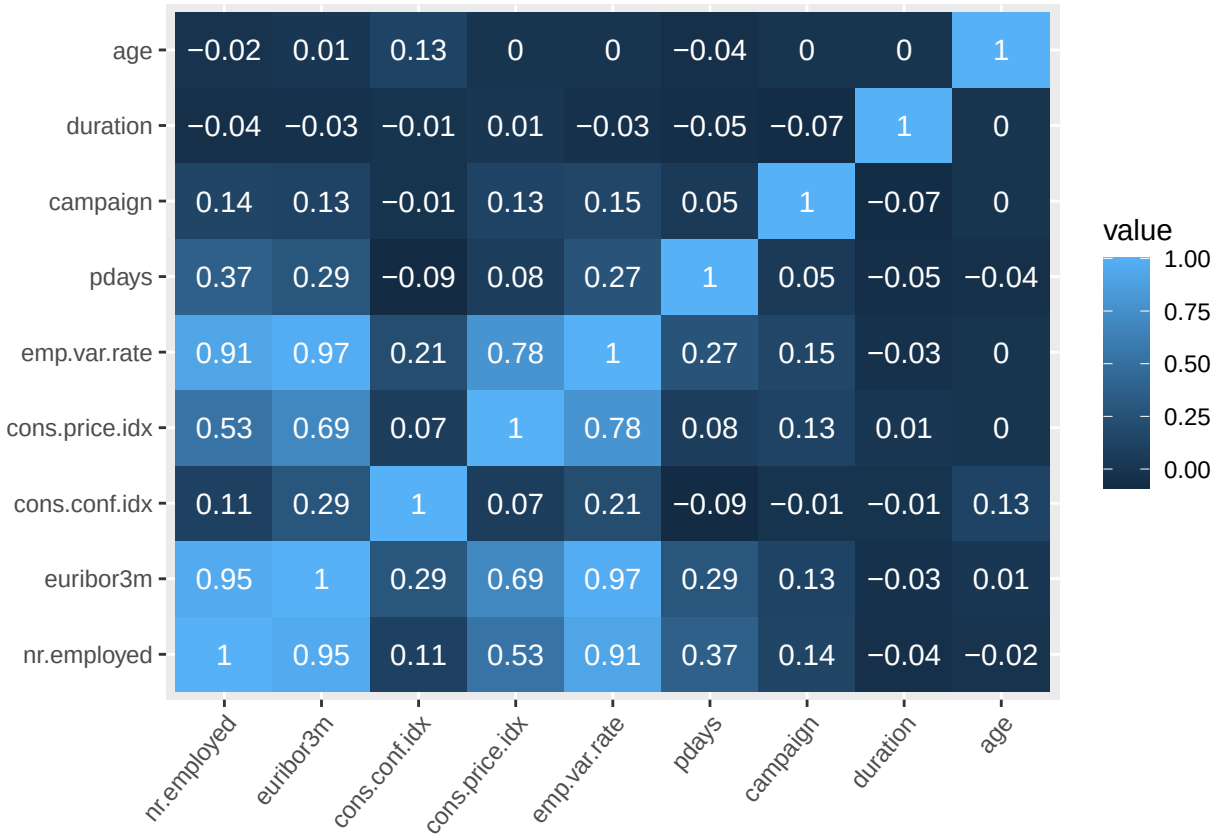

0
1
failure
nonexistent
success
57.69
49.88
42.07
34.26
26.45
18.64
10.83
3.02
-4.79
-12.6
-20.41



http://www.sthda.com/english/wiki/chi-square-test-of-independence-in-r https://www.mathsisfun.com/data/chi-square-test.html Positive residuals are in blue. Positive values in cells specify an attraction (positive association) between the corresponding row and column variables.
Negative residuals are in red. This implies a repulsion (negative association) between the corresponding row and column variables.

Continuous variables

```
col = c("nr.employed","euribor3m","cons.conf.idx","cons.price.idx","emp.var.rate","pdays","campaign","du

con = data[,col]
cor_matrix = round(cor(con),2)
melted_matrix <- melt(cor_matrix)

ggplot(melted_matrix, aes(x=Var1, y=Var2, fill= value)) +
  geom_tile()+
geom_text(aes(Var2, Var1, label = value), color = "white", size = 4) +
theme(
  axis.text.x=element_text(angle=50, hjust=1),
  axis.title.x = element_blank(),
  axis.title.y = element_blank(),
  legend.direction = "vertical")
```



euribor3m, emp.var.rate, nr.employed are highly correlated with each other. CPI is also highly correlated with these three >0.5.

Continuous and Categorical variables

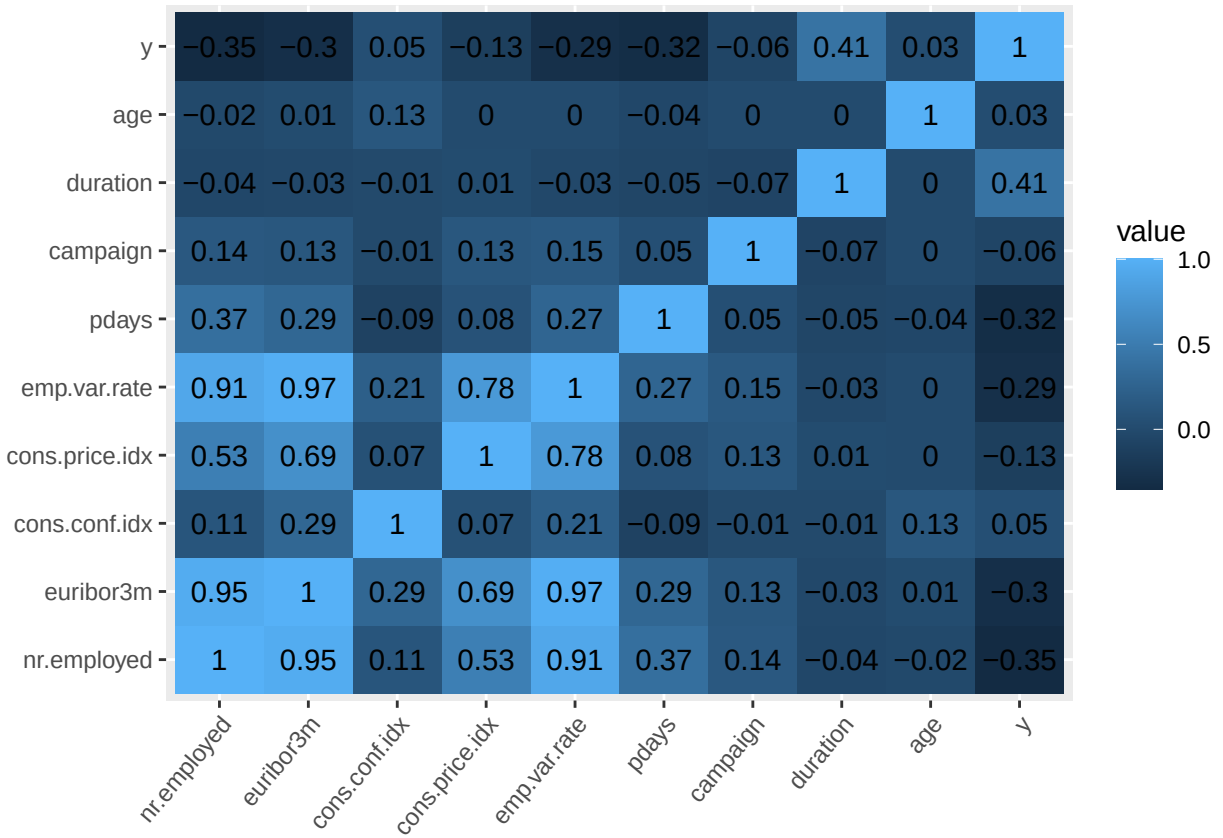
```
col = c("nr.employed","euribor3m","cons.conf.idx","cons.price.idx","emp.var.rate","pdays","campaign","du

results <- purrr::map(data[,col],~aov(.x~data$y))

m=1
for (i in col){

  pval <- unlist(summary(results[[m]]))
  print(paste("P value with ",i,"is",pval["Pr(>F)1"][[1]]))
  m=m+1

}
```

```
## [1] "P value with  nr.employed is 0"
## [1] "P value with  euribor3m is 0"
## [1] "P value with  cons.conf.idx is 6.7473008096419e-26"
## [1] "P value with  cons.price.idx is 1.54911167450653e-157"
## [1] "P value with  emp.var.rate is 0"
## [1] "P value with  pdays is 0"
## [1] "P value with  campaign is 5.38488043761305e-38"
```

```
## [1] "P value with  duration is 0"
## [1] "P value with  age is 4.98867350408353e-09"
```

Continuous and Categorical variables - using pearson. which is known as point- biserial since the response variable will be encoded

```
cor_matrix = round(cor(data1[,c(col,'y')]),2)
melted_matrix <- melt(cor_matrix)

ggplot(melted_matrix, aes(x=Var1, y=Var2, fill= value)) +
  geom_tile()+
geom_text(aes(Var2, Var1, label = value), color = "black", size = 4) +
theme(
  axis.text.x=element_text(angle=50, hjust=1),
  axis.title.x = element_blank(),
  axis.title.y = element_blank(),
  legend.direction = "vertical")
```



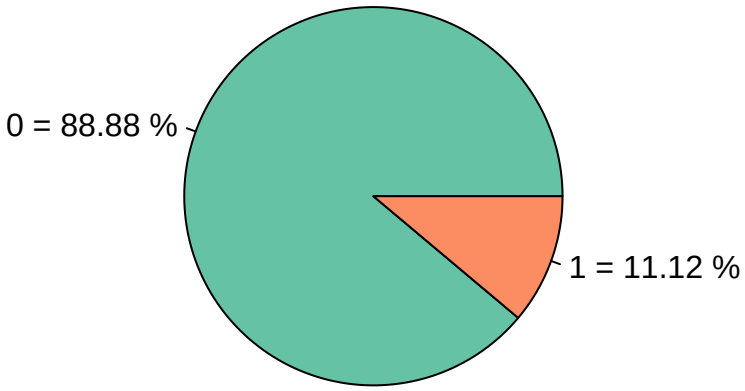
Duration can be considered. nr.emploed, cpi, emp.var.rate,euribor are dependent of eac other- So, nr.employed can be considered since it has relatively larger coefficient. (let's also try nr.eployed with cpi) p days can also be considered. Rest of the coeffiecients seem sma;ll

Unbalanced Data

```r
#install.packages("RColorBrewer")
library(RColorBrewer)

color <- brewer.pal(length(count), "Set2")
```

```
## Warning in brewer.pal(length(count), "Set2"): minimal value for n is 3, returning requested palette w
```

```r
pi <- data1 %>% group_by(y) %>% count()
pie(pi$n,labels=paste(pi$y,"=",round(100*pi$n/sum(pi$n),2),"%"),col=color)
```



The algorithm receives significantly more examples from one class, prompting it to be biased towards that particular class. It does not learn what makes the other class "different" and fails to understand the underlying patterns that allow us to distinguish classes.

To treat this we better proceed with synthetic data generation- SMOTE https://www.analyticsvidhya.com/blog/2016/03/practical-guide-deal-imbalanced-classification-problems/