Datacleaning.R

qianhuili

2019-09-25

```
#title: "Data Preparation & Summary Stats"
#author: "Qianhui Li"
```

```
setwd("/Users/qianhuili/Desktop/GitHub/AAE724/Script/Data_cleaning")

library(tidyr)
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(leaps)
library(glmnet)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

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

```
## Loading required package: foreach
```

```
## Loaded glmnet 2.0-18
```

```r
library(ggplot2)
library(gmodels)
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```r
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```r
library(ISLR)
library(tree)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
library(ROCR)
```

```
## Loading required package: gplots
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```r
library(rpart)
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following object is masked from 'package:gmodels':
##
##     ci
```

```
## The following object is masked from 'package:glmnet':
##
##     auc
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(corrplot)
library(lfe)
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
library(tidyverse)
```

```
## Registered S3 method overwritten by 'cli':
##   method     from
##   print.tree tree
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.2.1 ──
```

```
## ✔ tibble  2.1.3     ✔ purrr   0.3.2
## ✔ readr   1.3.1     ✔ stringr 1.4.0
## ✔ tibble  2.1.3     ✔ forcats 0.4.0
## ── Conflicts ──────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ purrr::accumulate()  masks foreach::accumulate()
## ✖ gridExtra::combine() masks dplyr::combine()
## ✖ Matrix::expand()     masks tidyr::expand()
## ✖ dplyr::filter()      masks stats::filter()
## ✖ dplyr::lag()         masks stats::lag()
## ✖ car::recode()        masks dplyr::recode()
## ✖ MASS::select()       masks dplyr::select()
## ✖ purrr::some()        masks car::some()
## ✖ purrr::when()        masks foreach::when()
```

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
library(RColorBrewer)
library(ggpubr)
```

```
## Loading required package: magrittr
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:purrr':
##
##     set_names
```

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

```
library(wesanderson)
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:MASS':
##
##     select
## The following object is masked from 'package:ggplot2':
## 
##     last_plot
```

```
## The following object is masked from 'package:stats':
## 
##     filter
```

```
## The following object is masked from 'package:graphics':
## 
##     layout
```

```
library(corrplot)
#=============================================

##Data Preparation

bankoriginal<-read.csv("bank_data.csv",header=TRUE, sep=";", na.strings=c("unknown","non-existent","999"))
bank<-na.omit(bankoriginal)
sum(is.na(bank))
```

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

```
#As indicated by the data contributor, the duration is not known before a call is performed.
#Also, after the end of the call y is obviously known.
#Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.

bank = bank %>%
  select(-duration)
summary(bank)
```

```
##       age                   job          marital   
##  Min.   :17.00   admin.     :430   divorced:129  
##  1st Qu.:30.00   technician :206   married :687  
##  Median :37.00   retired    :150   single  :494  
##  Mean   :41.51   blue-collar:108                 
##  3rd Qu.:51.00   student    :100                 
##  Max.   :88.00   management : 96                 
##                  (Other)    :220                 
##                education   default    housing    loan     
##  basic.4y           :124   no :1310   no :577   no :1112  
##  basic.6y           : 38   yes:   0   yes:733   yes: 198  
##  basic.9y           :107                                  
##  high.school        :310                                  
##  illiterate         :  0                                  
##  professional.course:184                                  
##  university.degree  :547                                  
##       contact         month     day_of_week    campaign    
##  cellular :1213   aug    :210   fri:221     Min.   :1.000  
##  telephone:  97   may    :207   mon:260     1st Qu.:1.000  
##                   nov    :173   thu:302     Median :1.000  
##                   sep    :136   tue:268     Mean   :1.824  
##                   jun    :135   wed:259     3rd Qu.:2.000  
##                   oct    :134               Max.   :8.000  
##                   (Other):315                              
##      pdays           previous           poutcome     emp.var.rate   
##  Min.   : 0.000   Min.   :1.00   failure    : 119   Min.   :-3.400  
##  1st Qu.: 3.000   1st Qu.:1.00   nonexistent:   0   1st Qu.:-2.900  
##  Median : 6.000   Median :1.00   success    :1191   Median :-1.800  
##  Mean   : 5.982   Mean   :1.65                      Mean   :-2.114  
##  3rd Qu.: 7.000   3rd Qu.:2.00                      3rd Qu.:-1.700  
##  Max.   :27.000   Max.   :7.00                      Max.   :-0.100  
##                                                                     
##  cons.price.idx  cons.conf.idx      euribor3m       nr.employed  
##  Min.   :92.20   Min.   :-50.80   Min.   :0.6340   Min.   :4964  
##  1st Qu.:92.65   1st Qu.:-42.00   1st Qu.:0.7200   1st Qu.:4992  
##  Median :93.08   Median :-38.30   Median :0.8790   Median :5018  
##  Mean   :93.34   Mean   :-38.29   Mean   :0.9839   Mean   :5029  
##  3rd Qu.:94.06   3rd Qu.:-31.40   3rd Qu.:1.0430   3rd Qu.:5076  
##  Max.   :94.77   Max.   :-26.90   Max.   :4.2860   Max.   :5196  
##                                                                  
##    y      
##  no :471  
##  yes:839  
##           
##           
##           
##           
## 
```

```
#convert variable types
sapply(bank,class)
```

```
##            age            job        marital      education        default 
##      "integer"       "factor"       "factor"       "factor"       "factor" 
##        housing           loan        contact          month    day_of_week 
##       "factor"       "factor"       "factor"       "factor"       "factor" 
##       campaign          pdays       previous       poutcome   emp.var.rate 
##      "integer"      "integer"      "integer"       "factor"      "numeric" 
## cons.price.idx  cons.conf.idx      euribor3m    nr.employed              y 
##      "numeric"      "numeric"      "numeric"      "numeric"       "factor" 
```

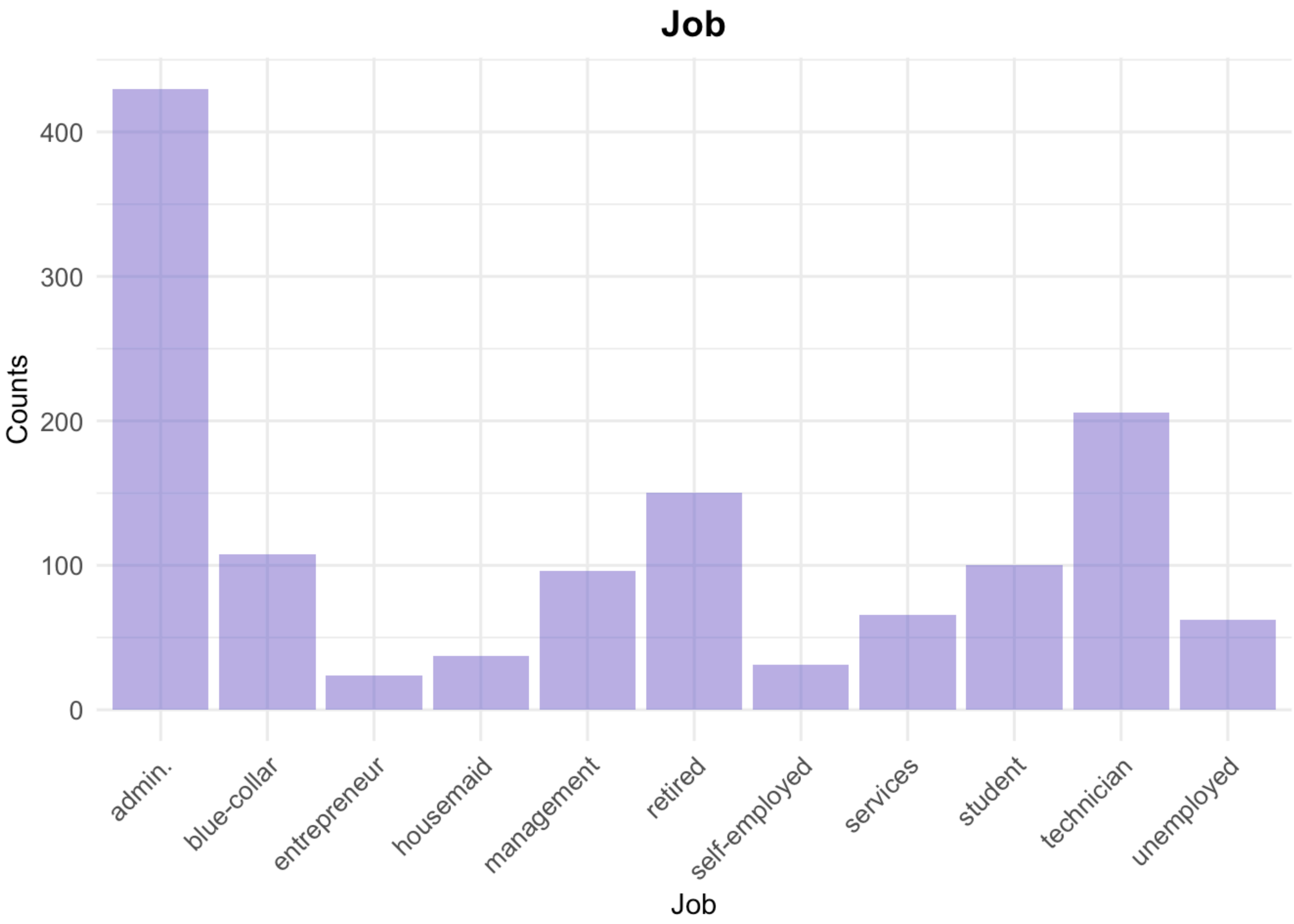
```
  #numerical variables
bank$age <- as.numeric(bank$age)
bank$campaign <- as.numeric(bank$campaign)
bank$pdays <- as.numeric(bank$pdays)
bank$previous <- as.numeric(bank$previous)
bank$emp.var.rate <- as.numeric(bank$emp.var.rate)
bank$cons.price.idx <- as.numeric(bank$cons.price.idx)
bank$cons.conf.idx <- as.numeric(bank$cons.conf.idx)
bank$euribor3m <- as.numeric(bank$euribor3m)
bank$nr.employed <- as.numeric(bank$nr.employed)

  #categorical variables
bank$job <-as.factor(bank$job)
bank$marital <-as.factor(bank$marital)
bank$education <-as.factor(bank$education)
bank$default <-as.factor(bank$default)
bank$loan <-as.factor(bank$loan)
bank$housing<-as.factor(bank$housing)
bank$contact <-as.factor(bank$contact)
bank$poutcome <-as.factor(bank$poutcome)
bank$day_of_week <-as.factor(bank$day_of_week)
bank$month <-as.factor(bank$month)

bank$y<-ifelse(bank$y =='yes',1,0)
bank$y <-as.factor(bank$y)



#=============================================

##Summary Statistics
summary(bank)
```

```
##       age                  job          marital
##  Min.   :17.00   admin.     :430   divorced:129
##  1st Qu.:30.00   technician :206   married :687
##  Median :37.00   retired    :150   single  :494
##  Mean   :41.51   blue-collar:108
##  3rd Qu.:51.00   student    :100
##  Max.   :88.00   management : 96
##                  (Other)    :220
##                education   default     housing     loan
##  basic.4y           :124   no :1310   no :577   no :1112
##  basic.6y           : 38   yes:   0   yes:733   yes: 198
##  basic.9y           :107
##  high.school        :310
##  illiterate         :  0
##  professional.course:184
##  university.degree  :547
##       contact          month       day_of_week    campaign
##  cellular :1213   aug    :210   fri:221     Min.   :1.000
##  telephone:  97   may    :207   mon:260     1st Qu.:1.000
##                   nov    :173   thu:302     Median :1.000
##                   sep    :136   tue:268     Mean   :1.824
##                   jun    :135   wed:259     3rd Qu.:2.000
##                   oct    :134               Max.   :8.000
##                   (Other):315
##      pdays           previous            poutcome      emp.var.rate
##  Min.   : 0.000   Min.   :1.00   failure    : 119   Min.   :-3.400
##  1st Qu.: 3.000   1st Qu.:1.00   nonexistent:   0   1st Qu.:-2.900
##  Median : 6.000   Median :1.00   success    :1191   Median :-1.800
##  Mean   : 5.982   Mean   :1.65                      Mean   :-2.114
##  3rd Qu.: 7.000   3rd Qu.:2.00                      3rd Qu.:-1.700
##  Max.   :27.000   Max.   :7.00                      Max.   :-0.100
##
##  cons.price.idx  cons.conf.idx     euribor3m      nr.employed   y
##  Min.   :92.20   Min.   :-50.80   Min.   :0.6340   Min.   :4964   0:471
##  1st Qu.:92.65   1st Qu.:-42.00   1st Qu.:0.7200   1st Qu.:4992   1:839
##  Median :93.08   Median :-38.30   Median :0.8790   Median :5018
##  Mean   :93.34   Mean   :-38.29   Mean   :0.9839   Mean   :5029
##  3rd Qu.:94.06   3rd Qu.:-31.40   3rd Qu.:1.0430   3rd Qu.:5076
##  Max.   :94.77   Max.   :-26.90   Max.   :4.2860   Max.   :5196
##
```

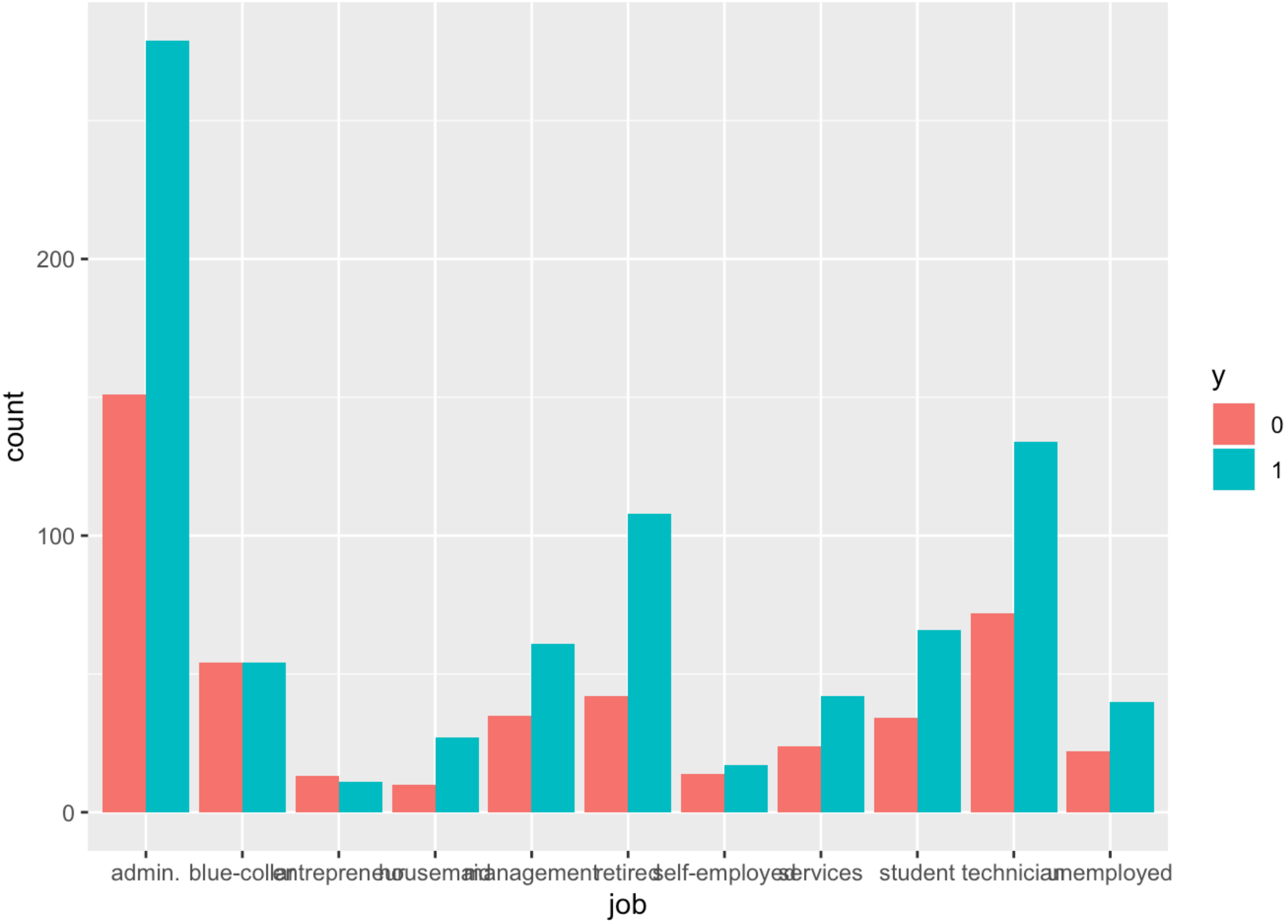
```
  #categorical variables exploration
pic_job <-ggplot(bank, aes(x=job)) + geom_histogram(aes(y=(..count..)), stat='count',
fill="slate blue", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "Job",
       x="Job", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_job
```



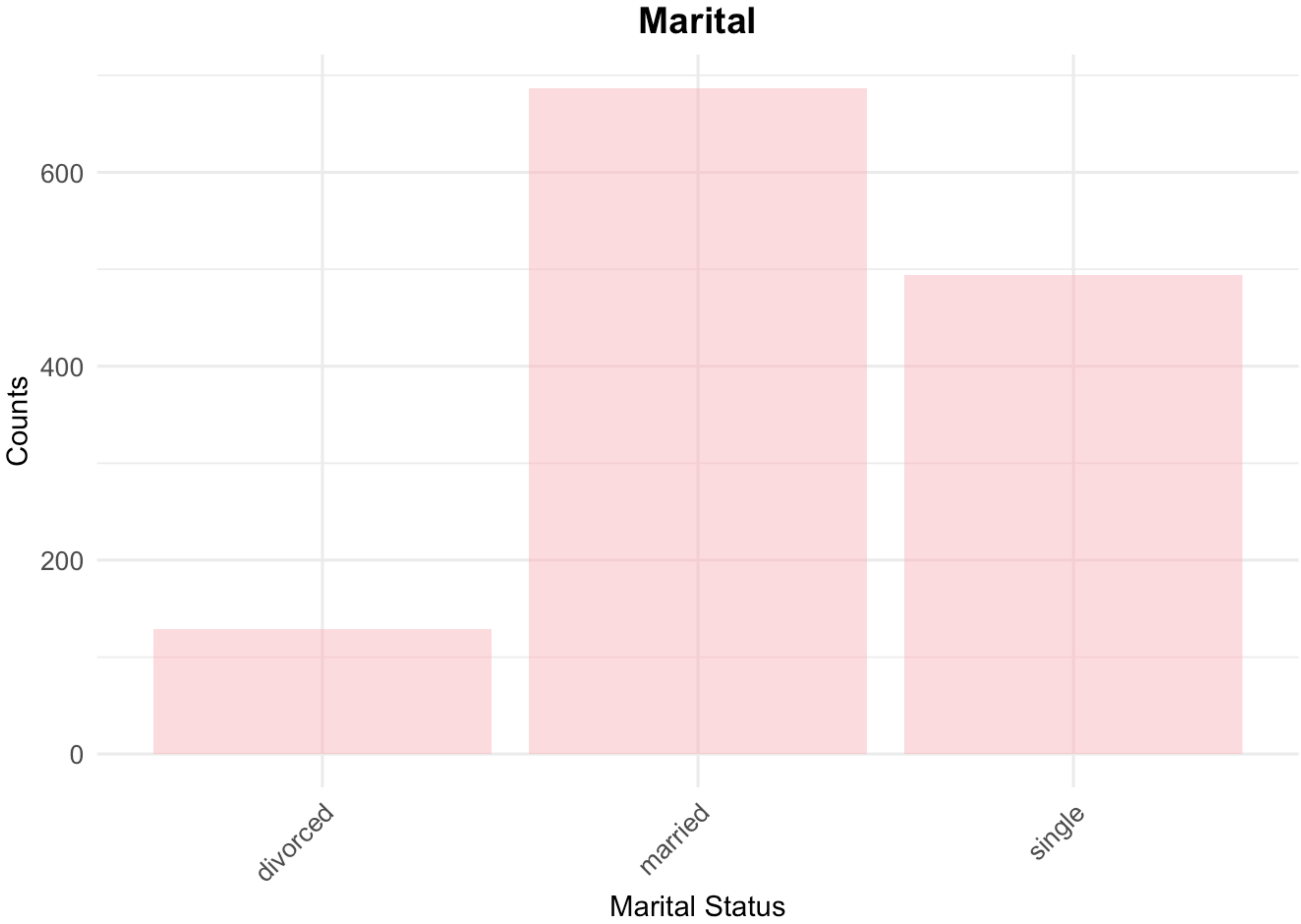
```
  #The graph shows that the there are alot of customers work in administritive sector
, and the least as entrepreneur.

aa <-ggplot(bank, aes(x = job , fill = y)) +
  geom_bar(stat='count', position='dodge')
aa
```



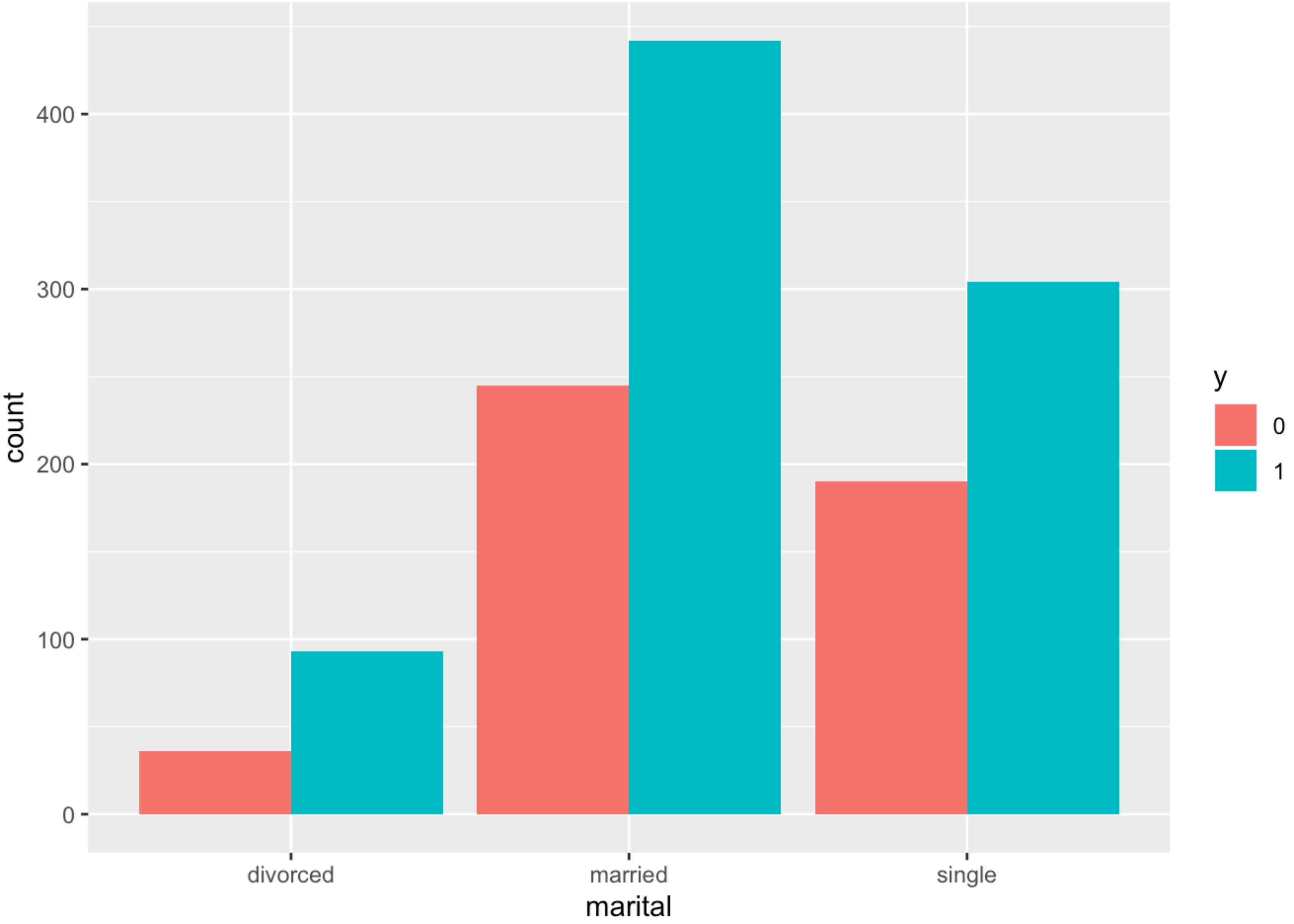
```
  #The graph shows that there are customers that are admin, retired, or technicial ar
e more willing to accept the offer.
#\\\\\\
pic_marital <-ggplot(bank, aes(x=marital)) + geom_histogram(aes(y=(..count..)), stat=
'count', fill="light pink", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "Marital",
       x="Marital Status", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_marital
```



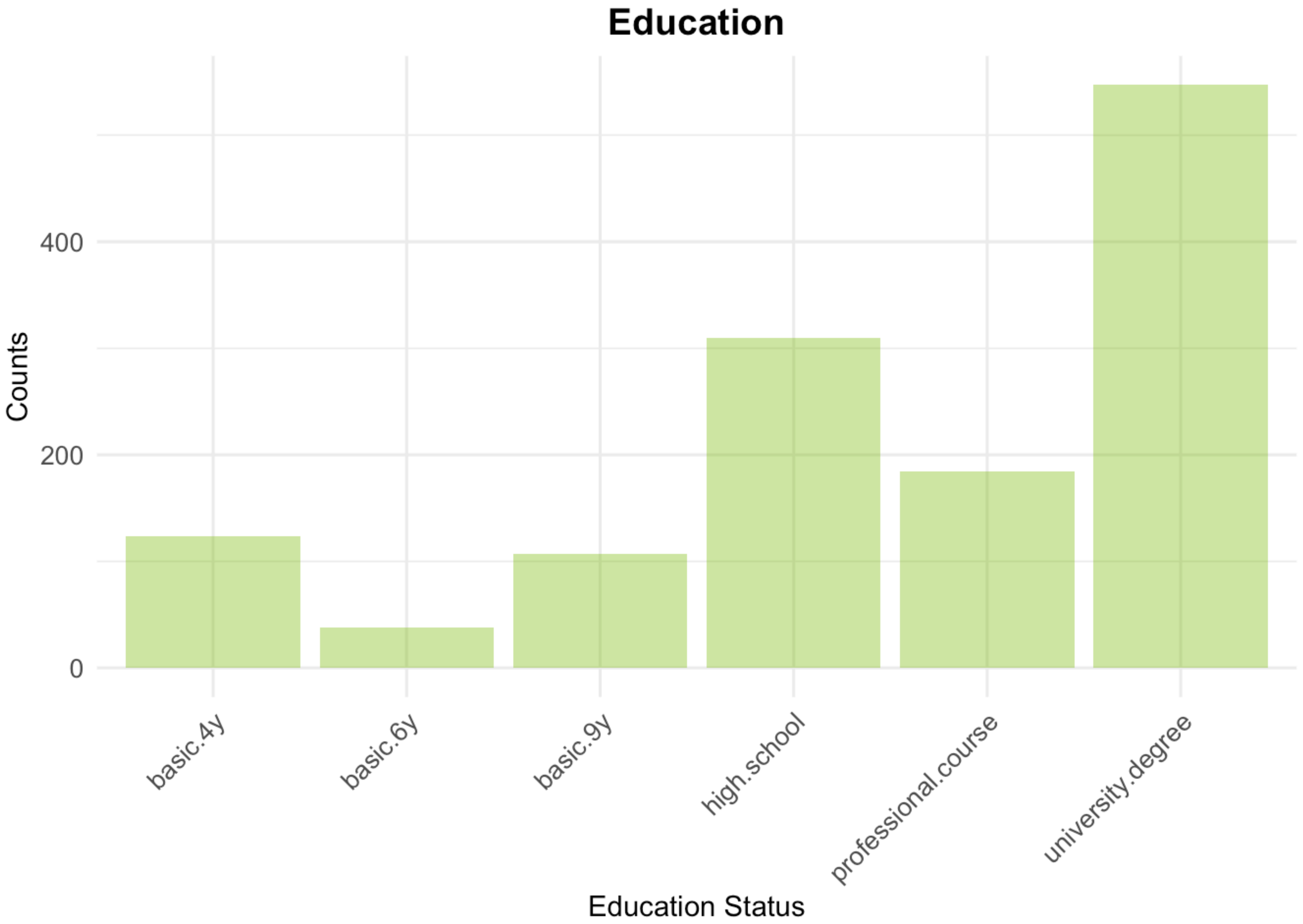
```
bb<-ggplot(bank, aes(x = marital , fill = y)) +
  geom_bar(stat='count', position='dodge')
bb
```



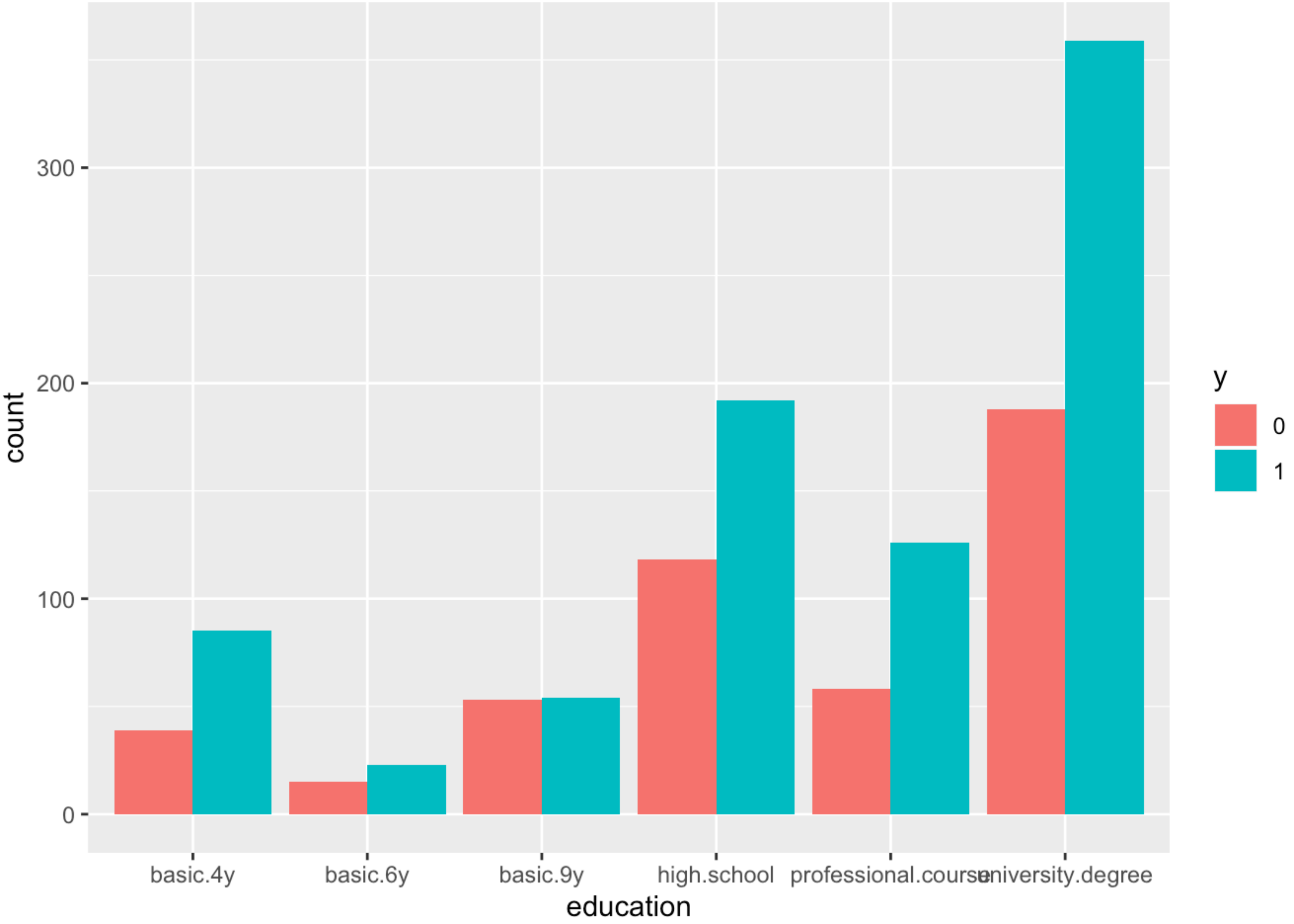
```
#\\\\\\

pic_edu <-ggplot(bank, aes(x=education)) + geom_histogram(aes(y=(..count..)), stat='c
ount', fill="yellowgreen", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "Education",
       x="Education Status", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_edu
```



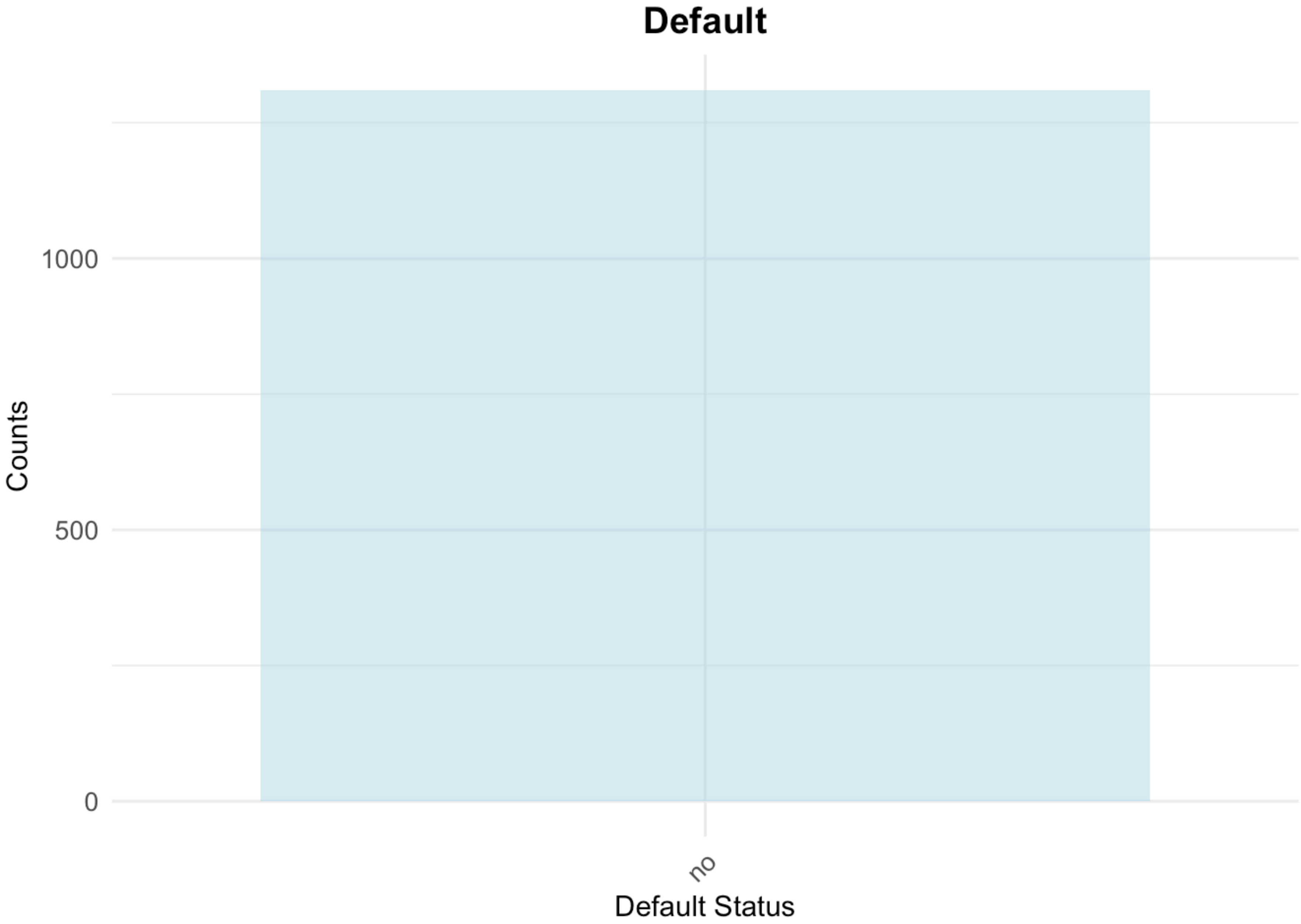
```
cc<-ggplot(bank, aes(x = education , fill = y)) +
  geom_bar(stat='count', position='dodge')
cc
```



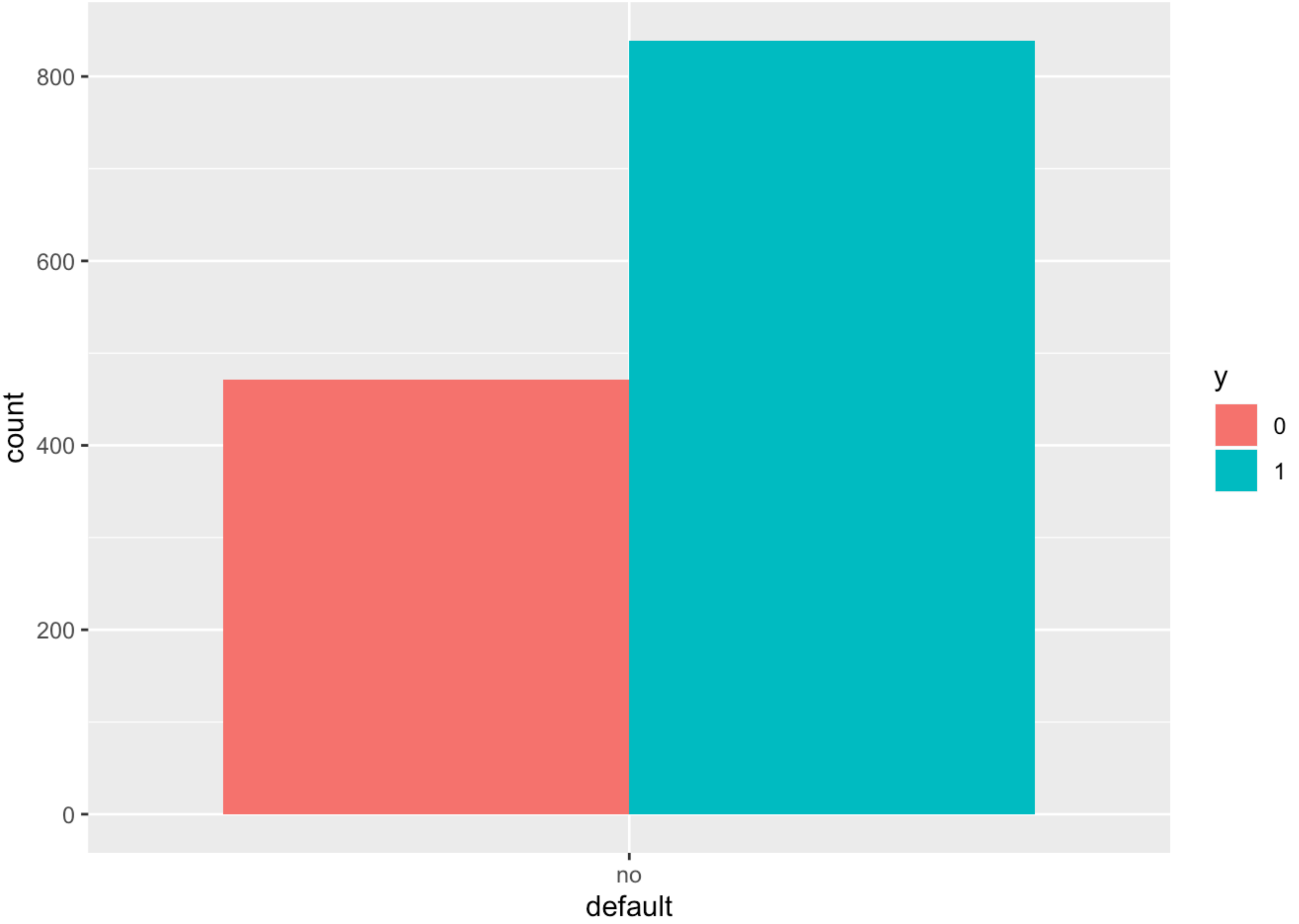
```
#\\\\\\

pic_default <-ggplot(bank, aes(x=default)) + geom_histogram(aes(y=(..count..)), stat=
'count', fill="light blue", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "Default",
       x="Default Status", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_default
```



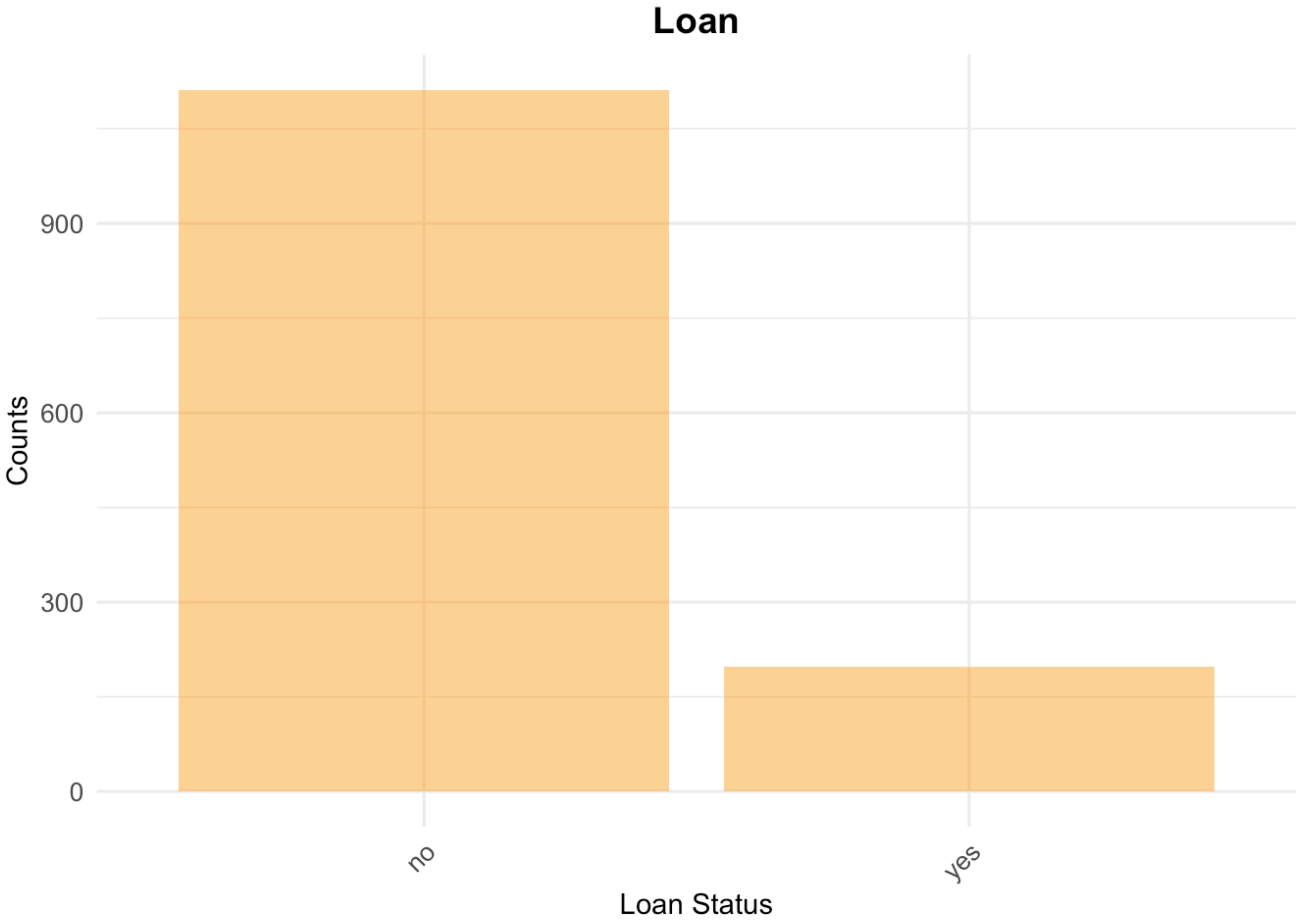
```
dd<-ggplot(bank, aes(x = default , fill = y)) +
  geom_bar(stat='count', position='dodge')
dd
```



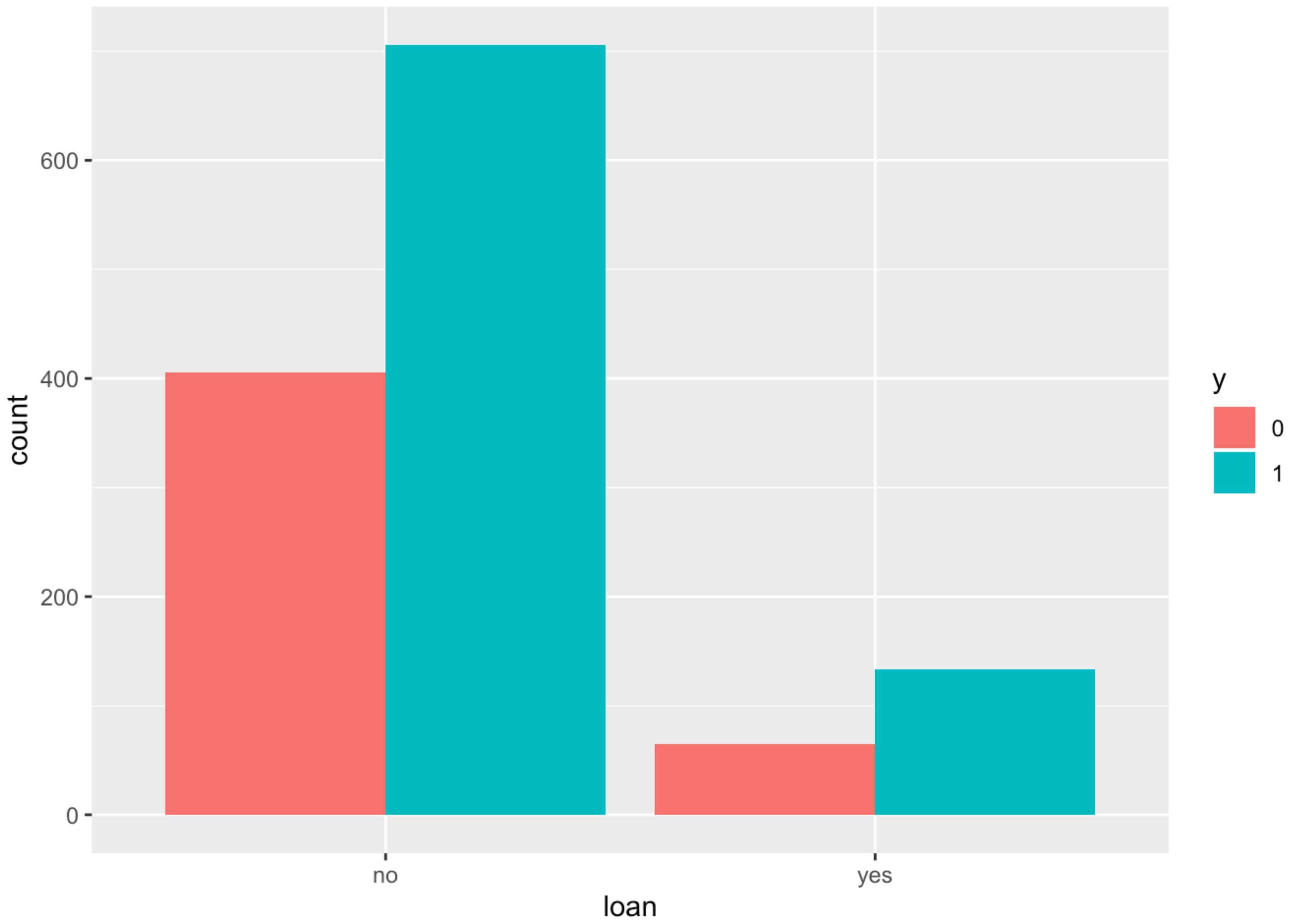
```
#\\\\\\

pic_loan <-ggplot(bank, aes(x=loan)) + geom_histogram(aes(y=(..count..)), stat='count
', fill="orange1", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "Loan",
       x="Loan Status", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_loan
```



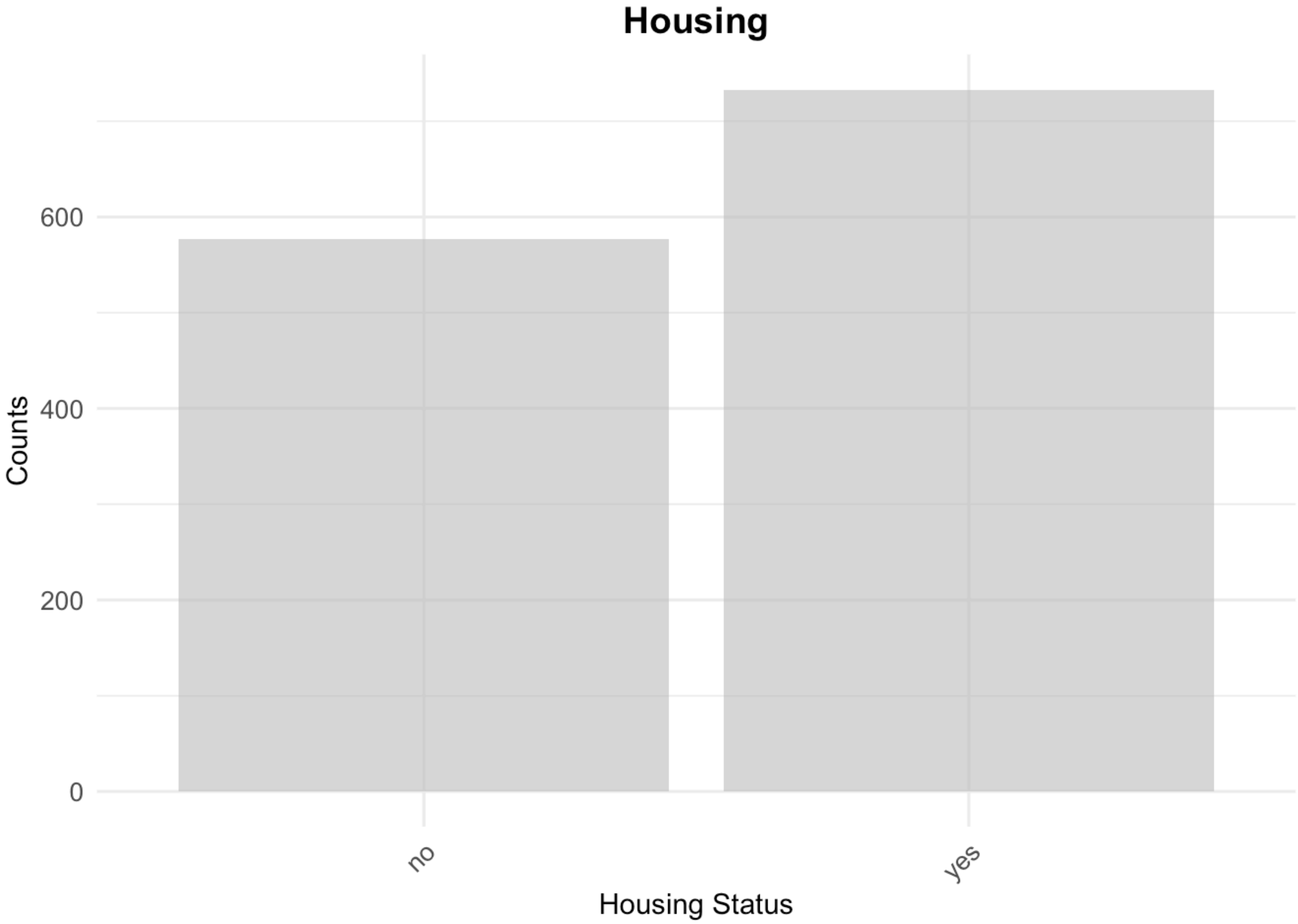
```
ee<-ggplot(bank, aes(x = loan , fill = y)) +
  geom_bar(stat='count', position='dodge')
ee
```



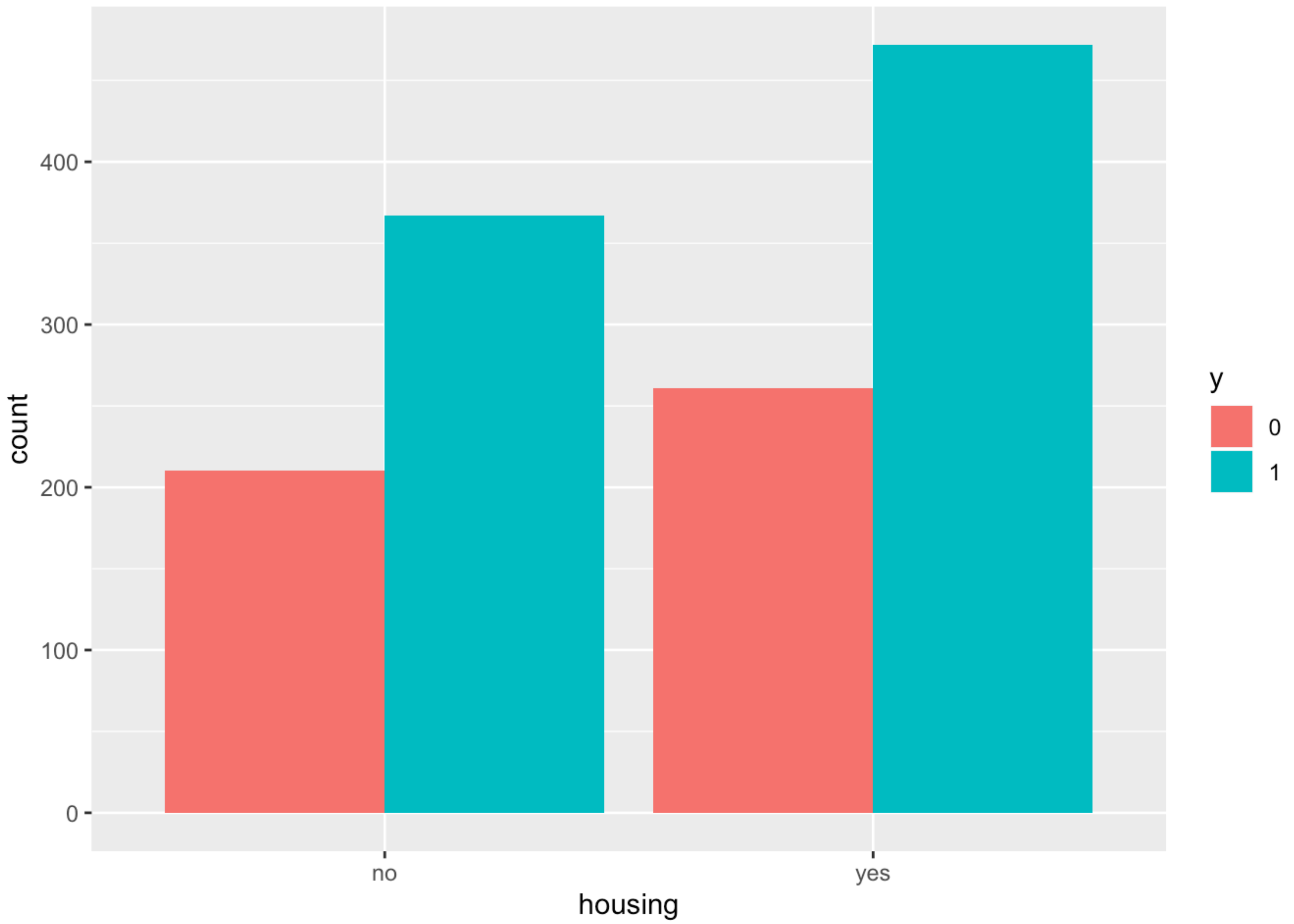
```
#\\\\\\
pic_housing <-ggplot(bank, aes(x=housing)) + geom_histogram(aes(y=(..count..)), stat=
'count', fill="grey69", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "Housing",
       x="Housing Status", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_housing
```



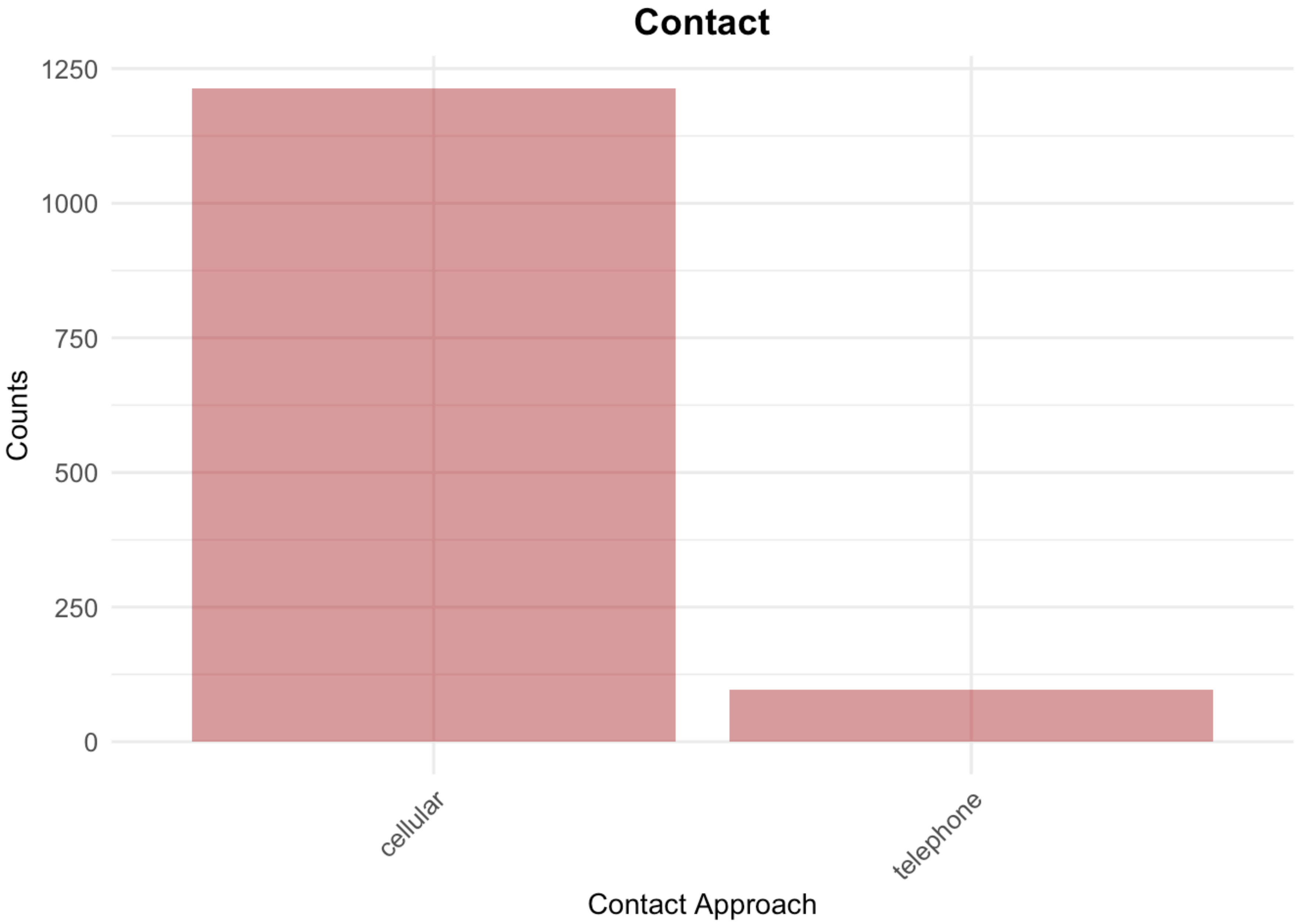
```
ff<-ggplot(bank, aes(x = housing , fill = y)) +
  geom_bar(stat='count', position='dodge')
ff
```



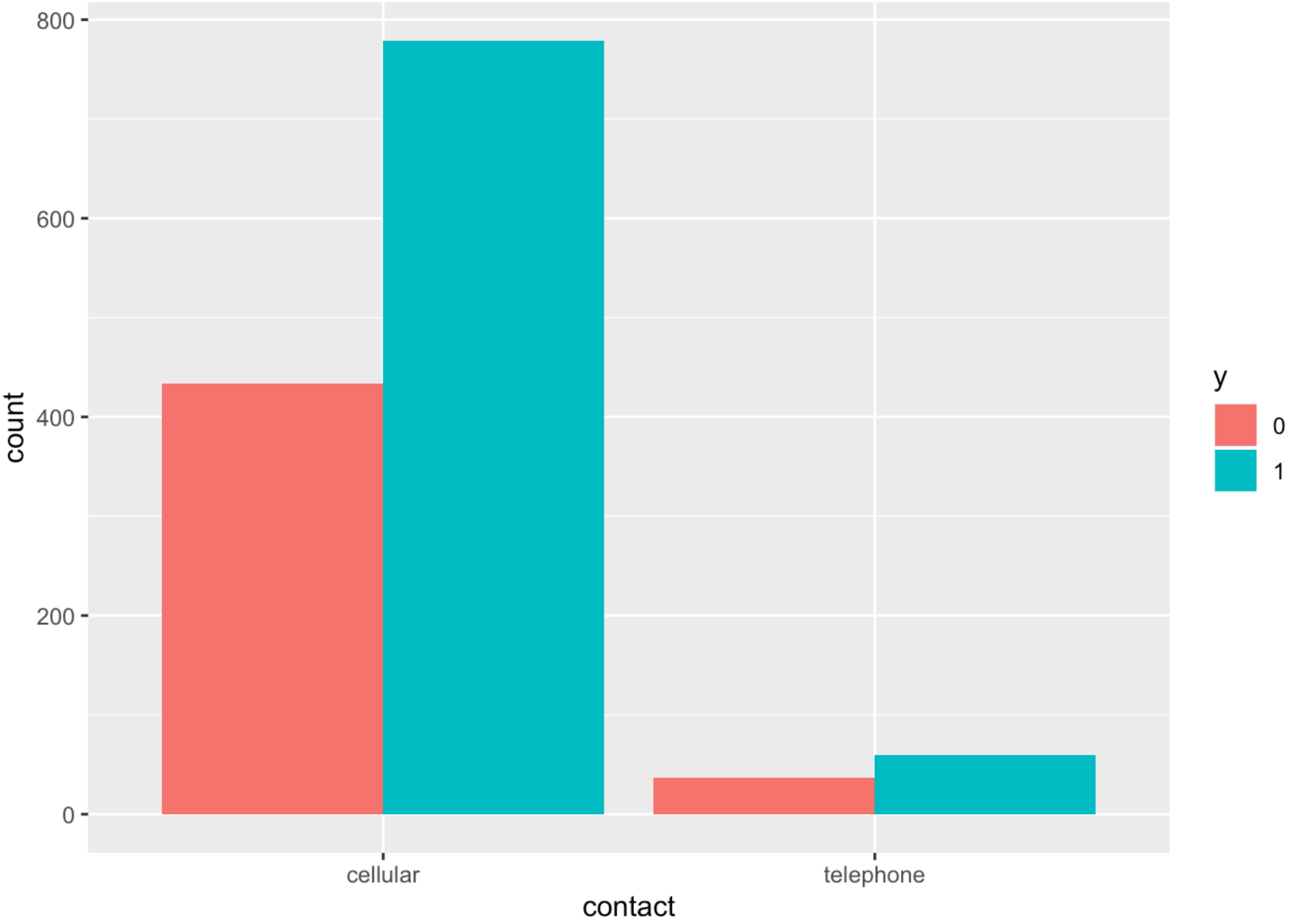
```
#\\\\\\
pic_contact <-ggplot(bank, aes(x=contact)) + geom_histogram(aes(y=(..count..)), stat=
'count', fill="firebrick", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "Contact",
       x="Contact Approach", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_contact
```



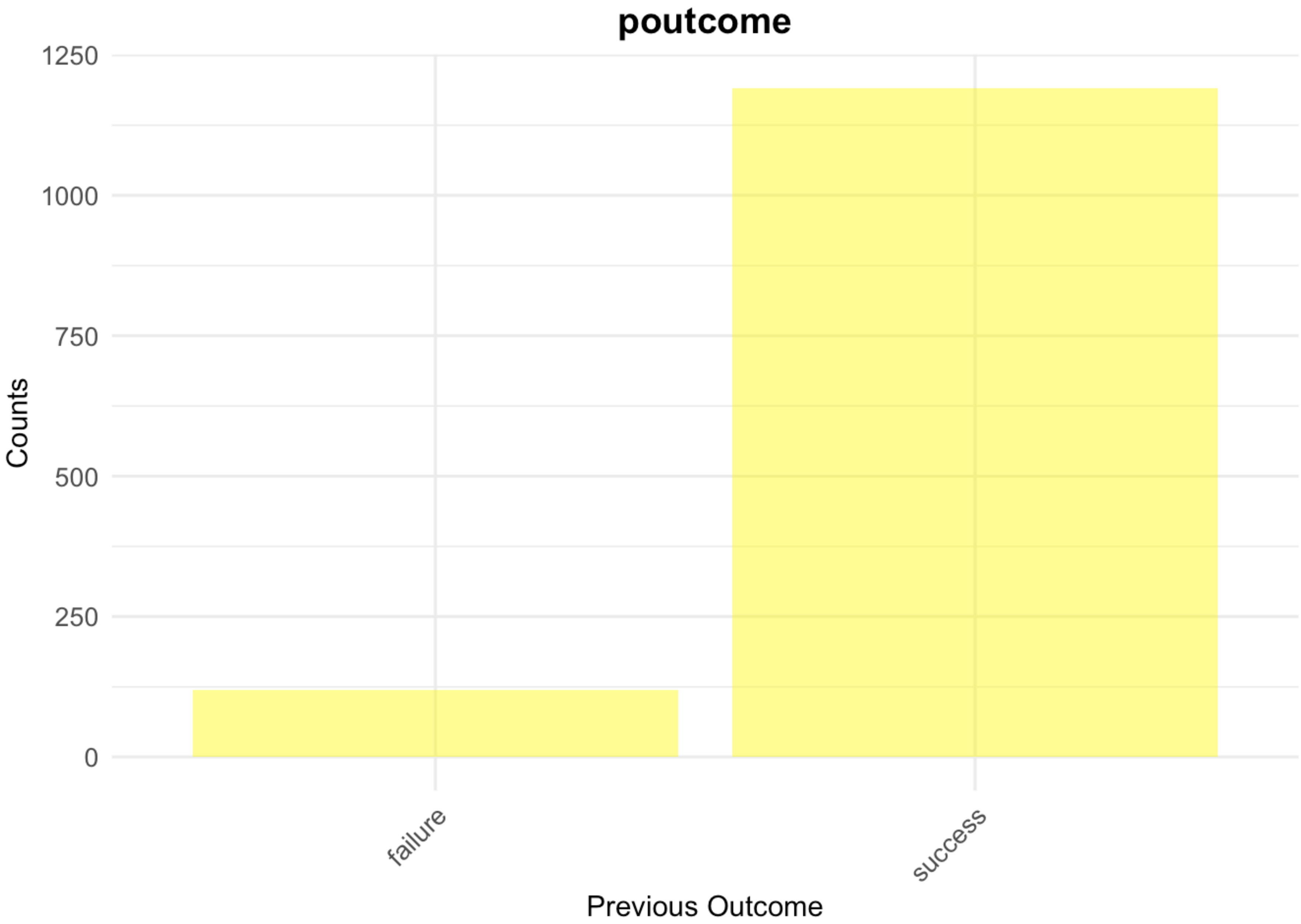
```
gg<-ggplot(bank, aes(x = contact , fill = y)) +
  geom_bar(stat='count', position='dodge')
gg
```



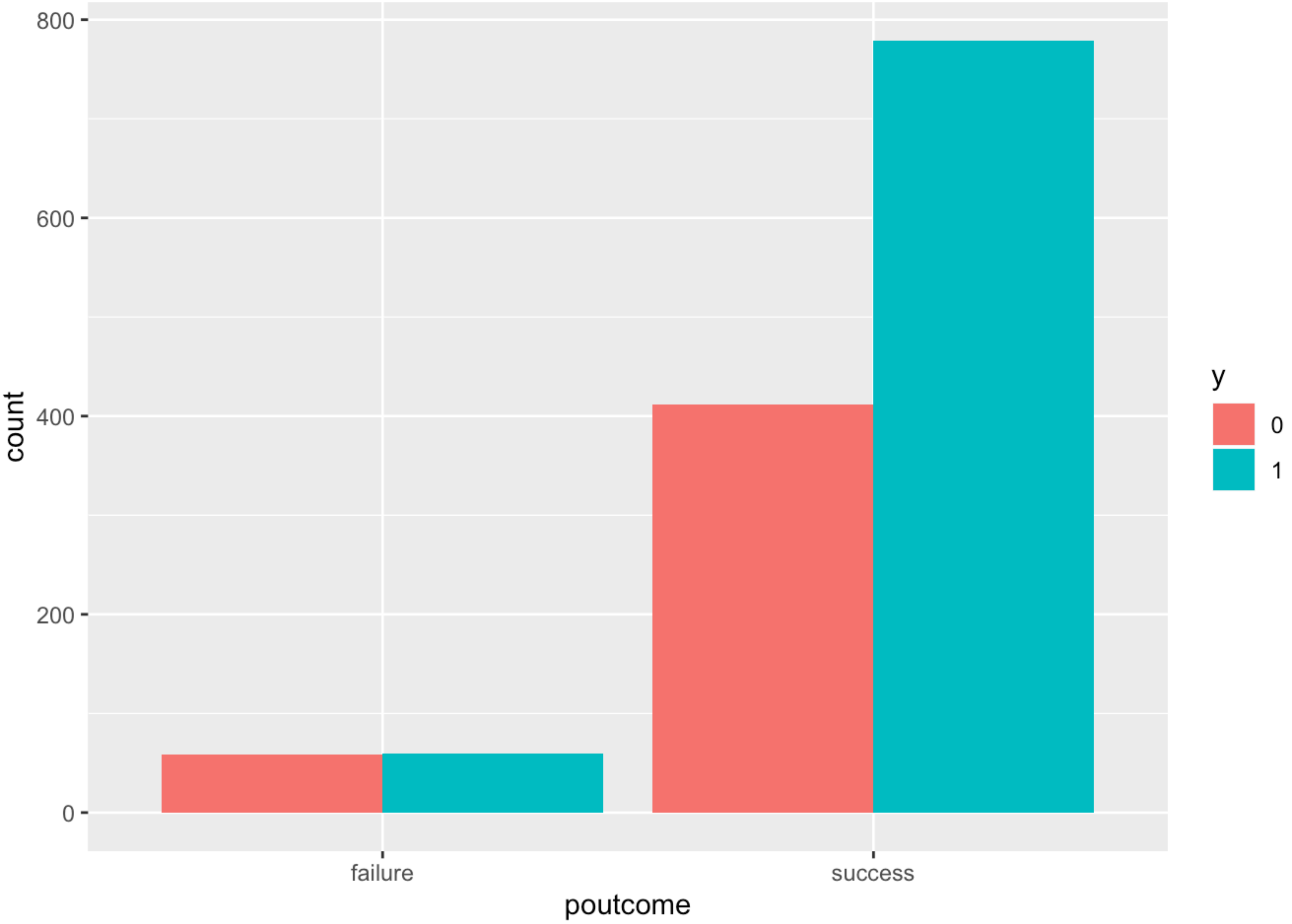
```
#\\\\\\
pic_poutcome <-ggplot(bank, aes(x=poutcome)) + geom_histogram(aes(y=(..count..)), sta
t='count', fill="yellow1", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "poutcome",
       x="Previous Outcome", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_poutcome
```



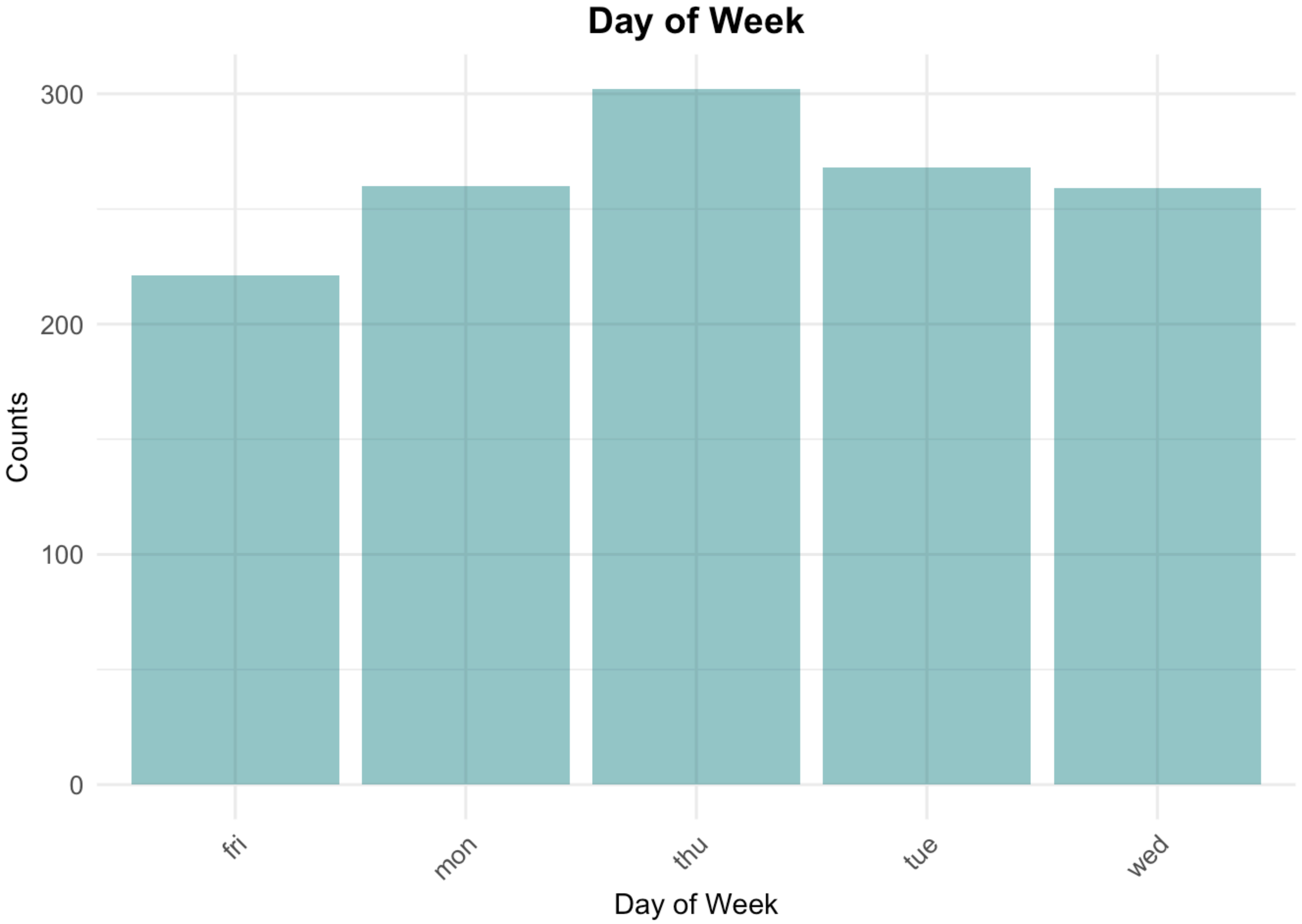
```
hh<-ggplot(bank, aes(x = poutcome , fill = y)) +
  geom_bar(stat='count', position='dodge')
hh
```



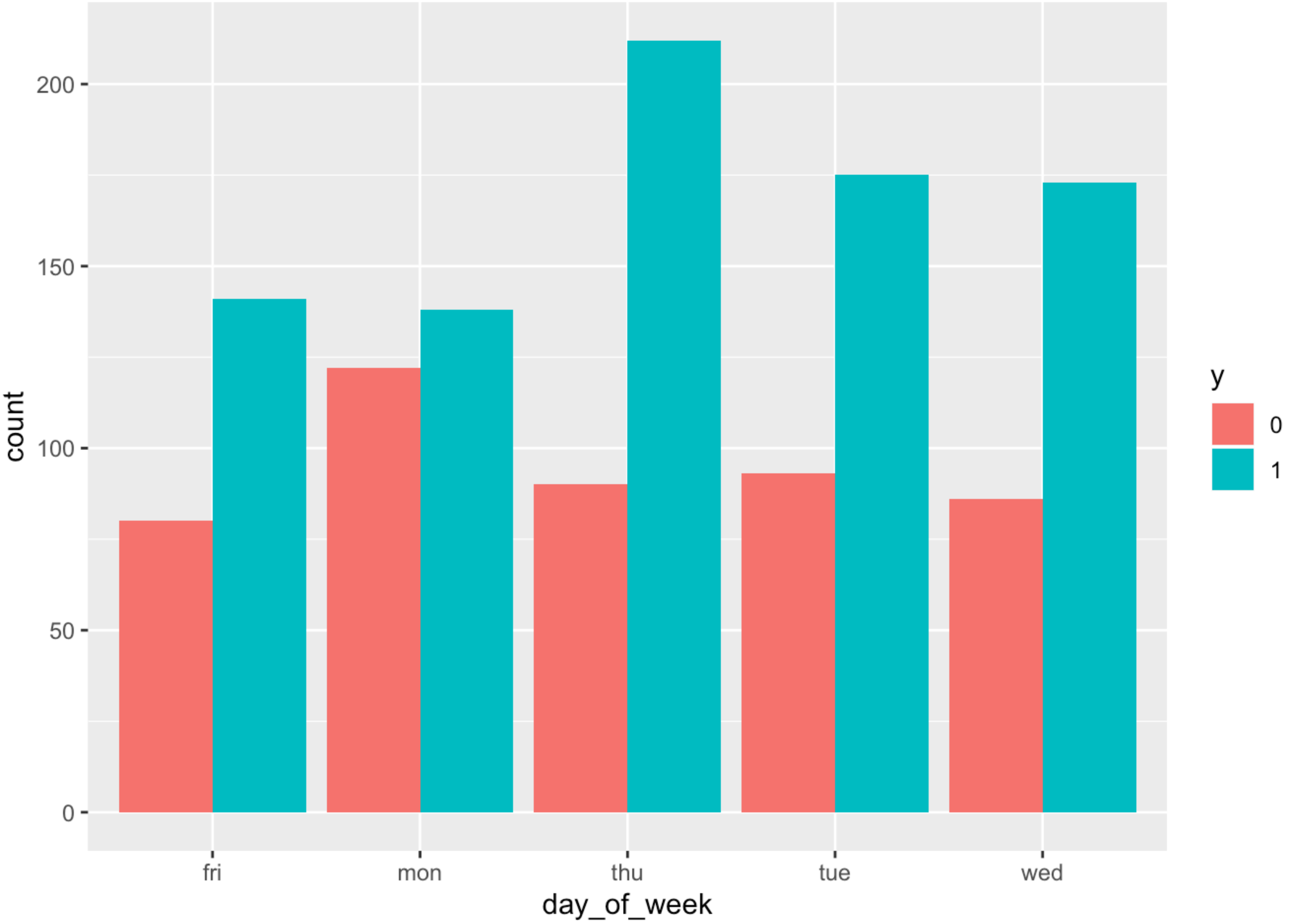
```
#\\\\\\
pic_dow <-ggplot(bank, aes(x=day_of_week)) + geom_histogram(aes(y=(..count..)), stat=
'count', fill="turquoise4", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title     = "Day of Week",
       x="Day of Week", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_dow
```



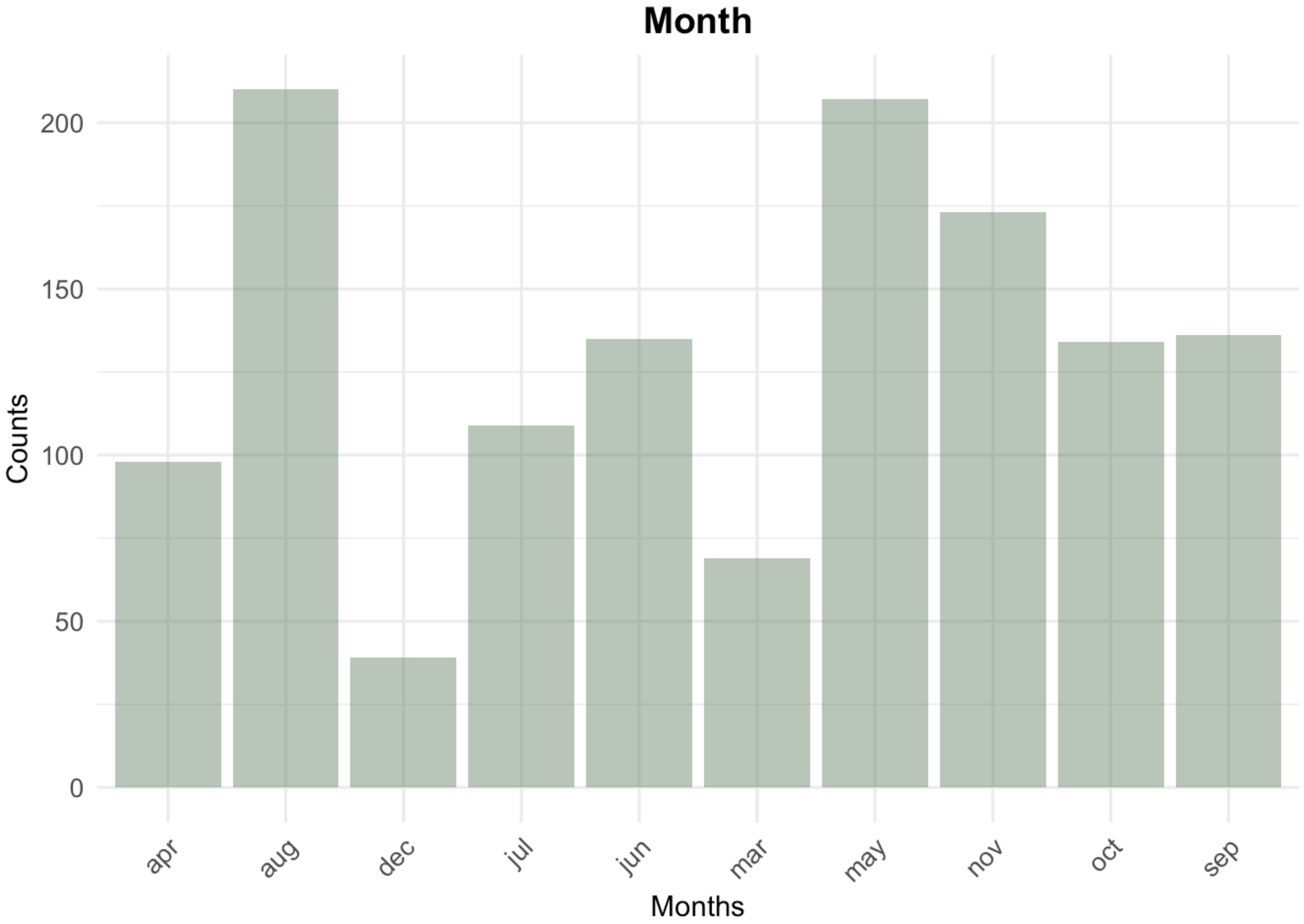
```
jj<-ggplot(bank, aes(x = day_of_week , fill = y)) +
  geom_bar(stat='count', position='dodge')
jj
```



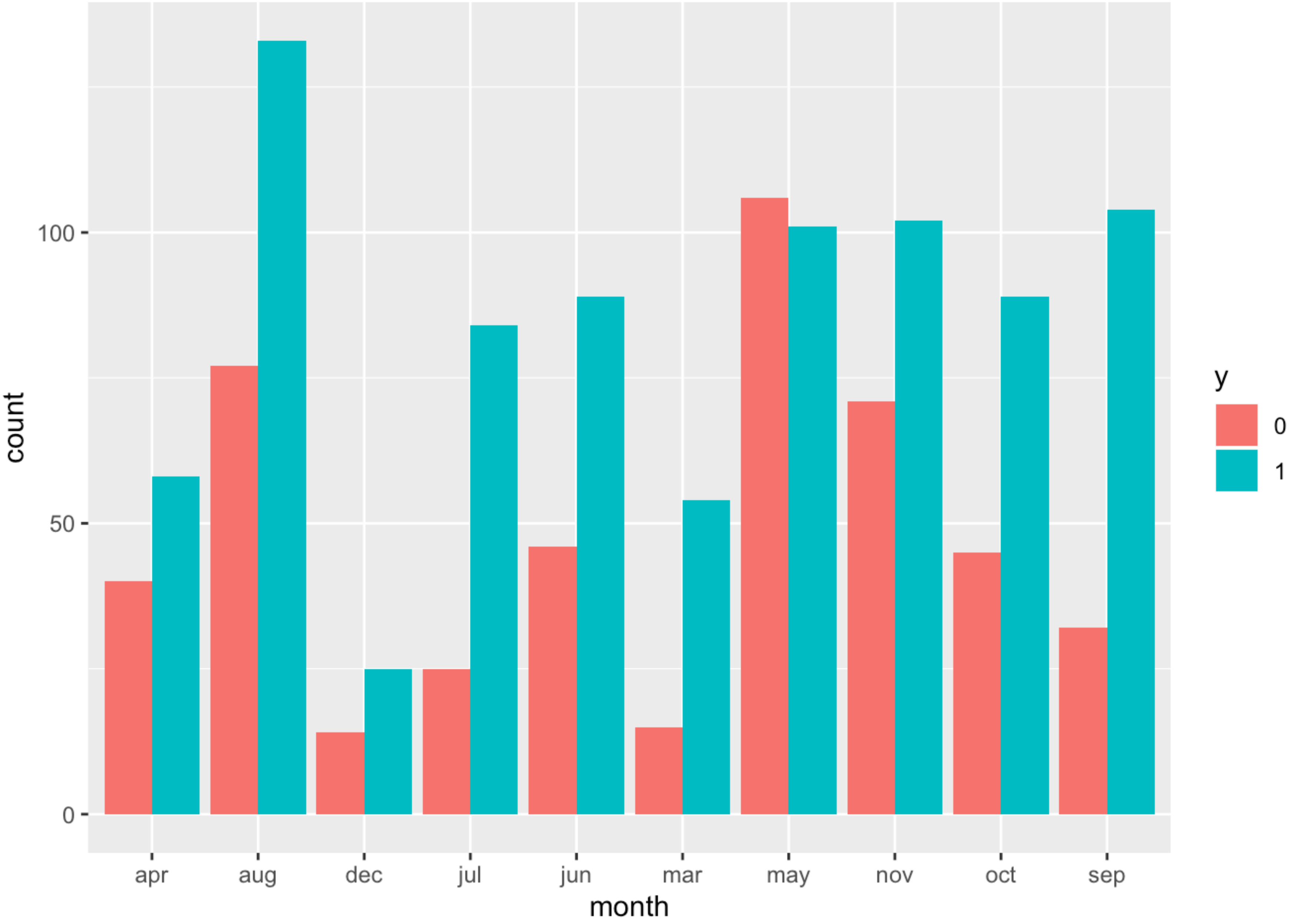
```
#\\\\\\
pic_month <-ggplot(bank, aes(x=month)) + geom_histogram(aes(y=(..count..)), stat='cou
nt', fill="darkseagreen4", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "Month",
       x="Months", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
pic_month
```



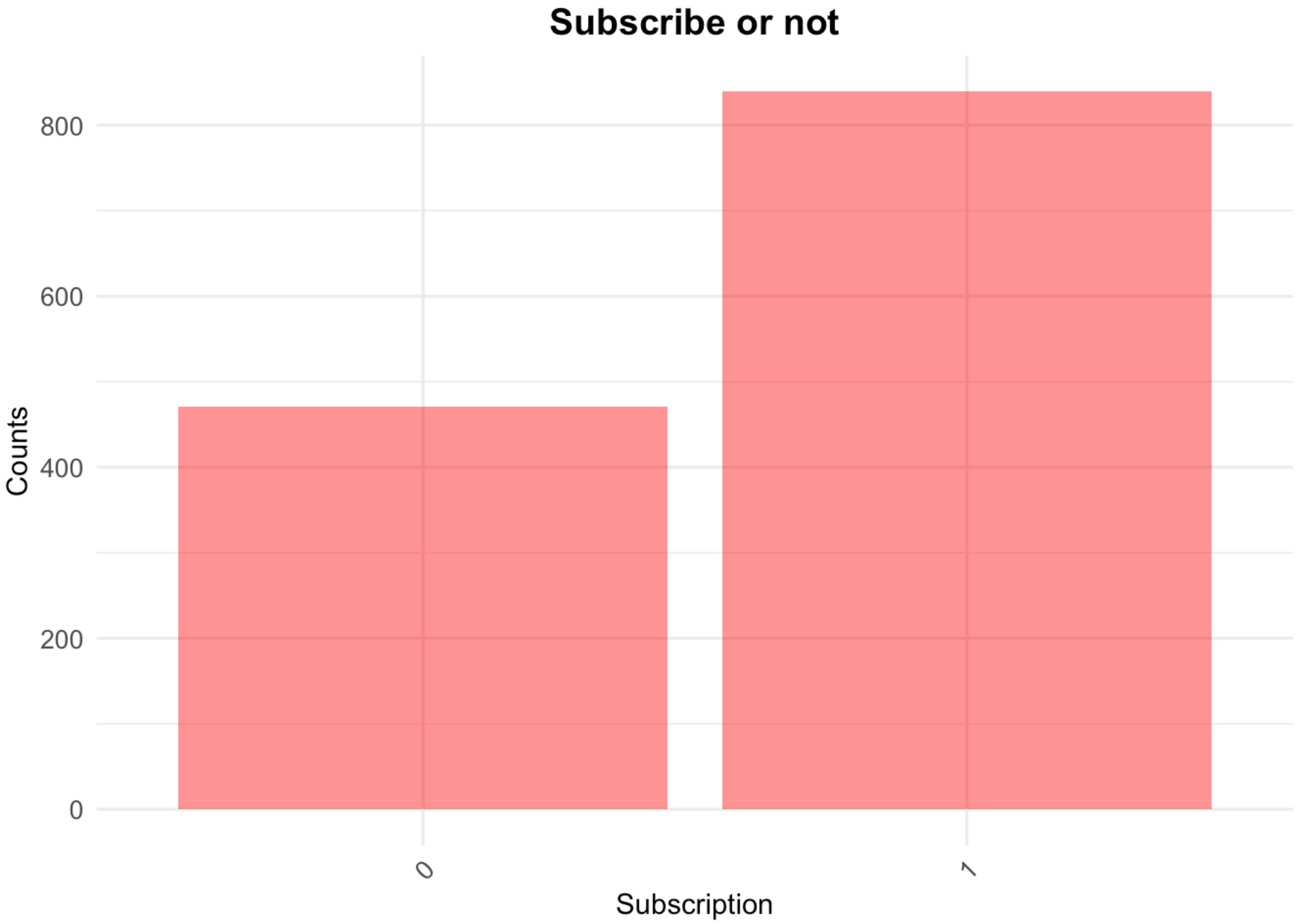
```
kk<-ggplot(bank, aes(x = month , fill = y)) +
  geom_bar(stat='count', position='dodge')
kk
```



```
#\\\\\\
#response variable
pic_y <-ggplot(bank, aes(x=y)) + geom_histogram(aes(y=(..count..)), stat='count', fil
l="red", alpha=0.5) + theme_minimal() +
  theme(plot.title    = element_text(face = "bold", size = 14, hjust = 0.5),
        axis.text.x     = element_text(angle = 45, hjust = 1, size=10),
        axis.text.y     = element_text(size=10)) +
  labs(title    = "Subscribe or not",
       x="Subscription", y="Counts")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

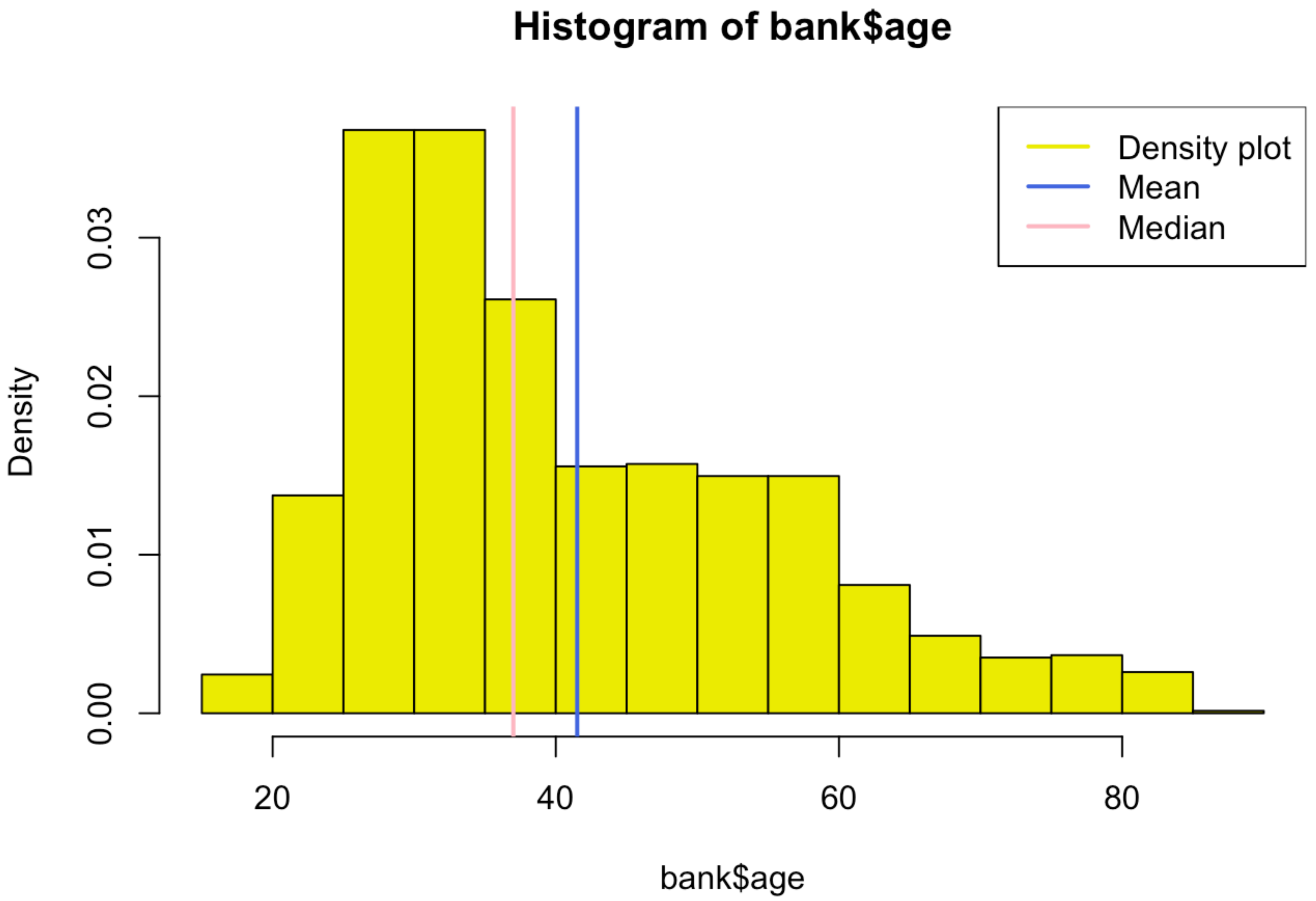
```
pic_y
```



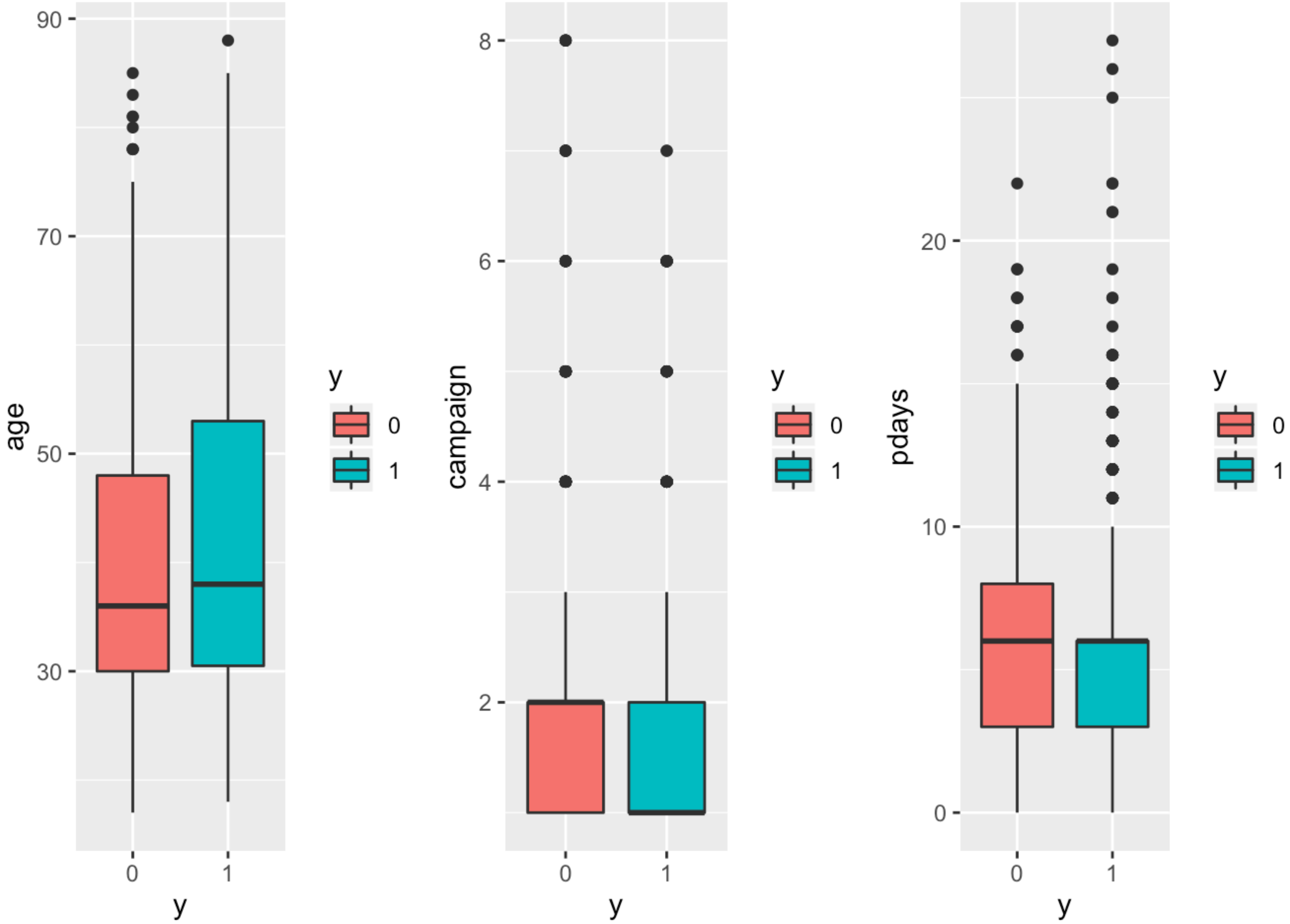
```
CrossTable(bank$y)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  1310
##
##
##           |         0 |         1 |
##           |-----------|-----------|
##           |       471 |       839 |
##           |     0.360 |     0.640 |
##           |-----------|-----------|
##
##
##
##
```

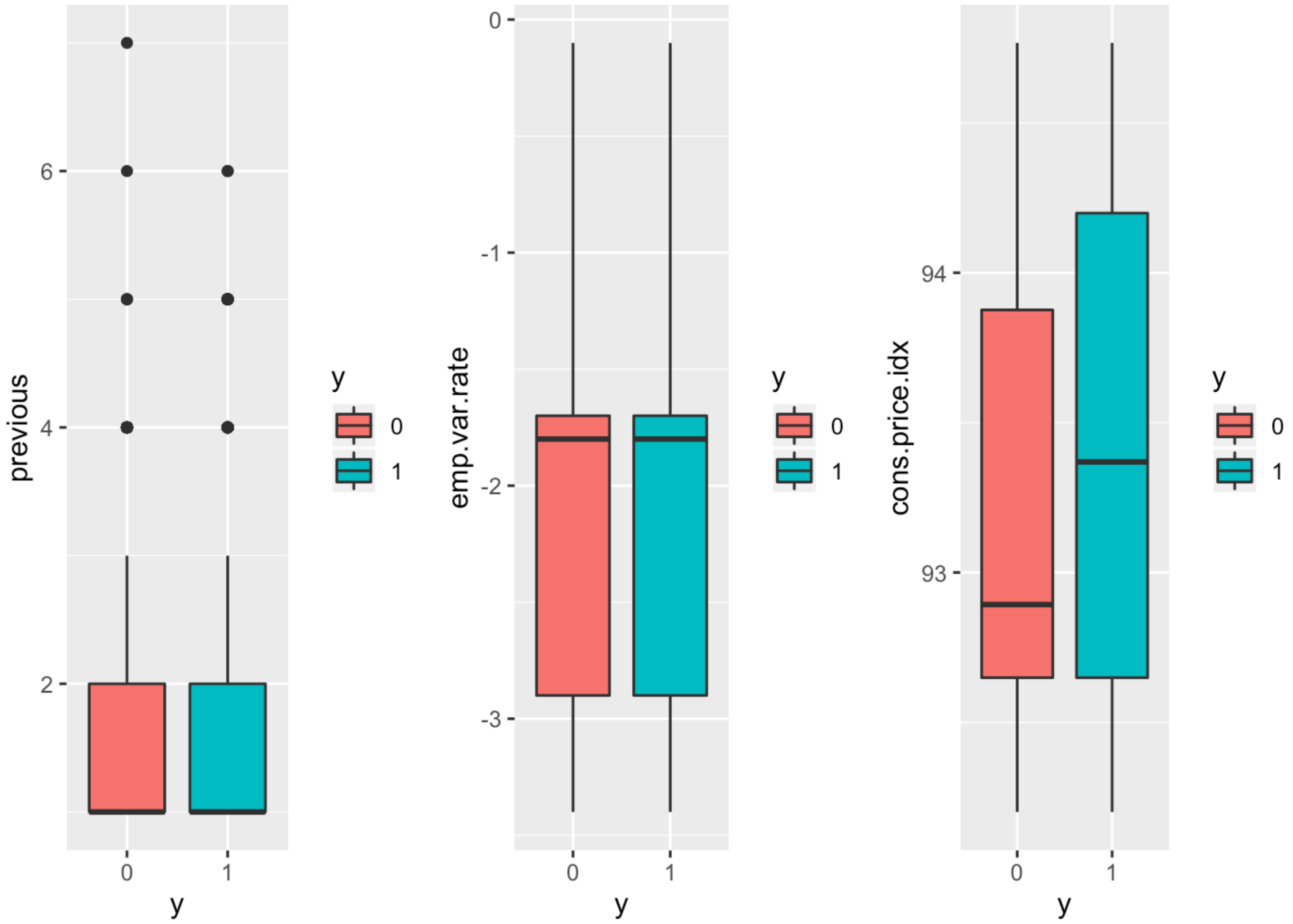
```
  #numerical variables exploration
p_age <- ggplot(bank, aes(y, age)) + geom_boxplot(aes(fill = y))
hist(bank$age, col = "yellow2", freq = FALSE)
abline(v = mean(bank$age),
       col = "royalblue",
       lwd = 2)
abline(v = median(bank$age),
       col = "light pink",
       lwd = 2)
legend(x = "topright",
       c("Density plot", "Mean", "Median"),
       col = c("yellow2", "royalblue", "light pink"),
       lwd = c(2, 2, 2))
```


Histogram of bank$age
Density plot
Mean
Median
Density
0.00
0.01
0.02
0.03
20
40
60
80
bank$age

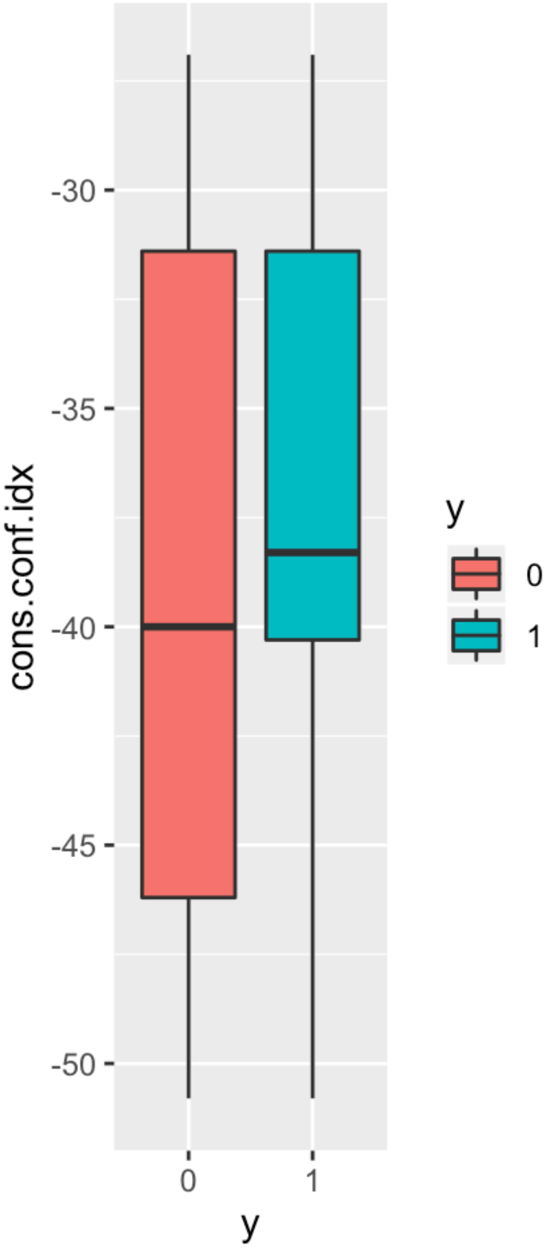
```
#The distribution shows that most customers oberved are less than 40 years old.


p_campaign <- ggplot(bank, aes(y, campaign)) + geom_boxplot(aes(fill = y))


p_pdays <- ggplot(bank, aes(y, pdays)) + geom_boxplot(aes(fill = y))


p_previous <- ggplot(bank, aes(y, previous)) + geom_boxplot(aes(fill = y))


p_emp.var.rate <- ggplot(bank, aes(y, emp.var.rate)) + geom_boxplot(aes(fill = y))


p_cons.price.idx <- ggplot(bank, aes(y, cons.price.idx)) + geom_boxplot(aes(fill = y)
)


p_cons.conf.idx<- ggplot(bank, aes(y, cons.conf.idx)) + geom_boxplot(aes(fill = y))


p_euribor3m<- ggplot(bank, aes(y, euribor3m)) + geom_boxplot(aes(fill = y))


p_nr.employed<- ggplot(bank, aes(y, nr.employed)) + geom_boxplot(aes(fill = y))


a <- c(p_age,p_campaign,p_pdays)
ggarrange(p_age,p_campaign,p_pdays,
          nrow = 1)
```

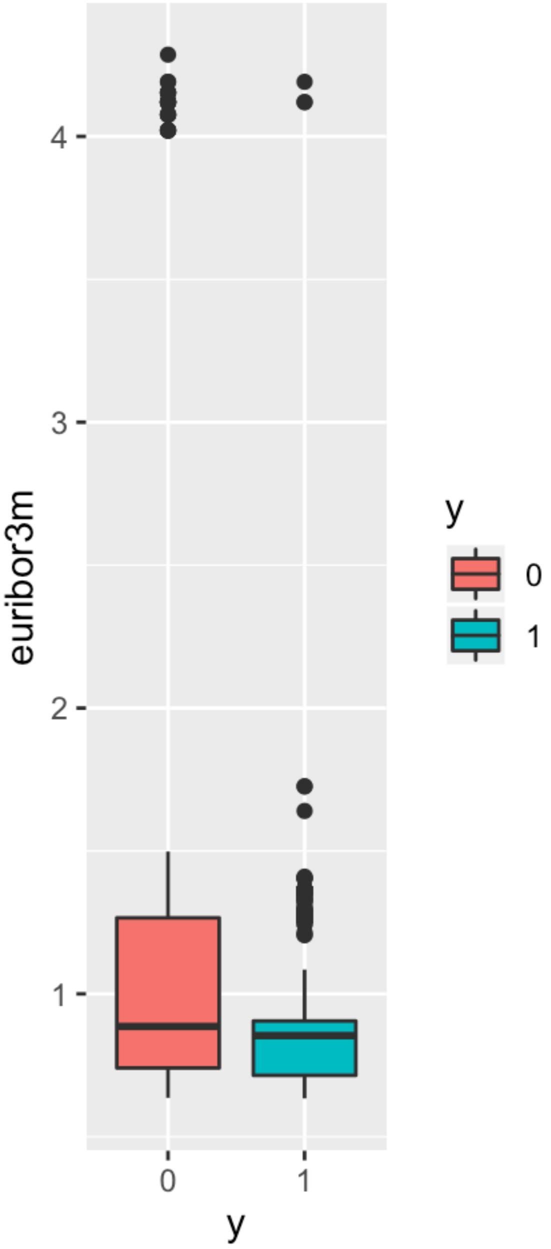


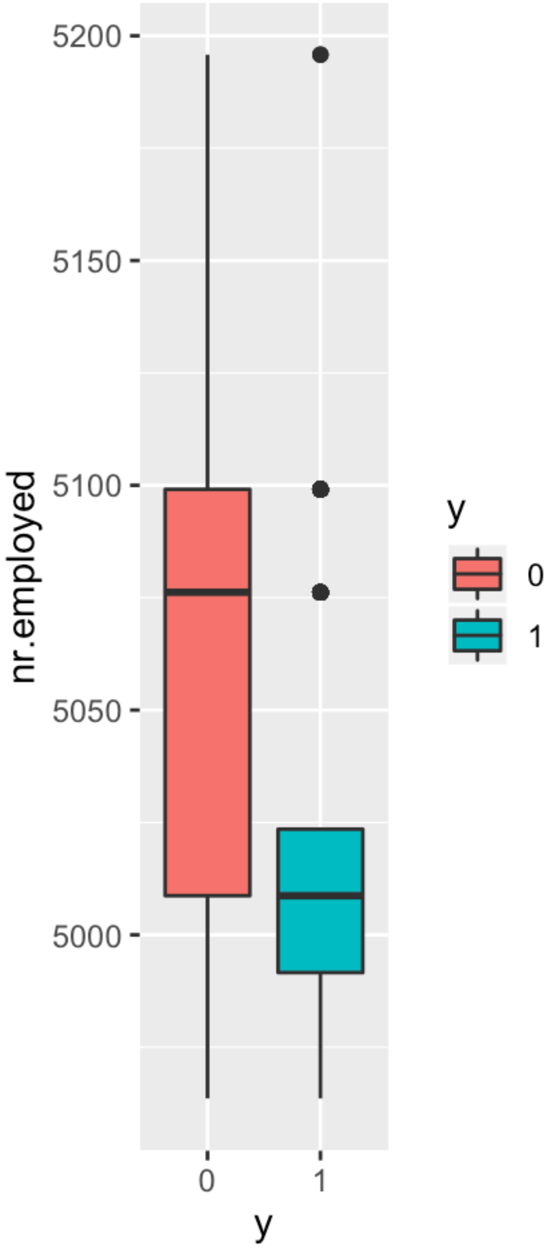
```
b <- c(p_previous,p_emp.var.rate,
       p_cons.price.idx)
ggarrange(p_previous,p_emp.var.rate,
          p_cons.price.idx,
          nrow = 1)
```



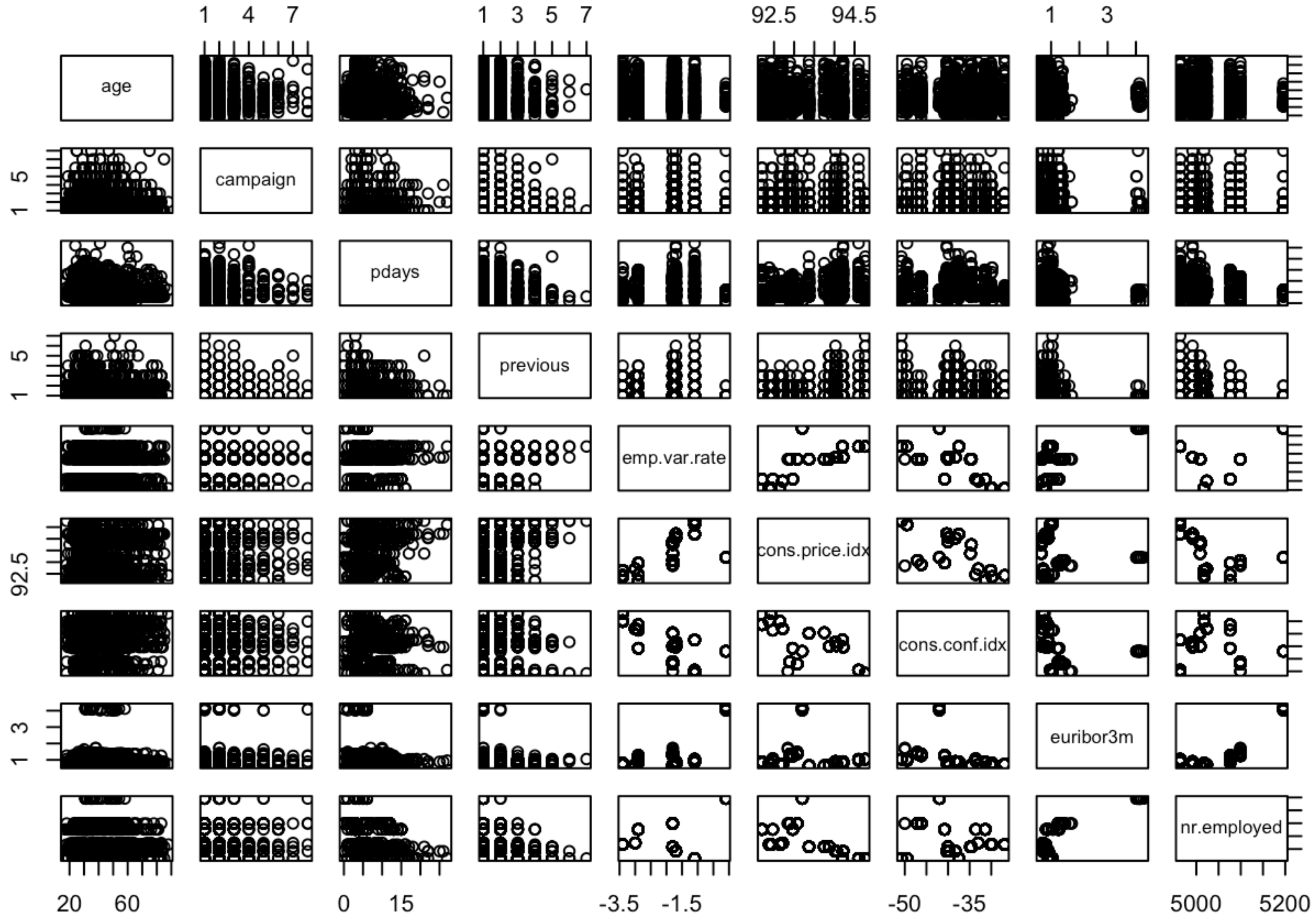
```
g <- c(p_cons.conf.idx,
       p_euribor3m,p_nr.employed)
ggarrange(p_cons.conf.idx,p_euribor3m,p_nr.employed,
          nrow = 1)
```



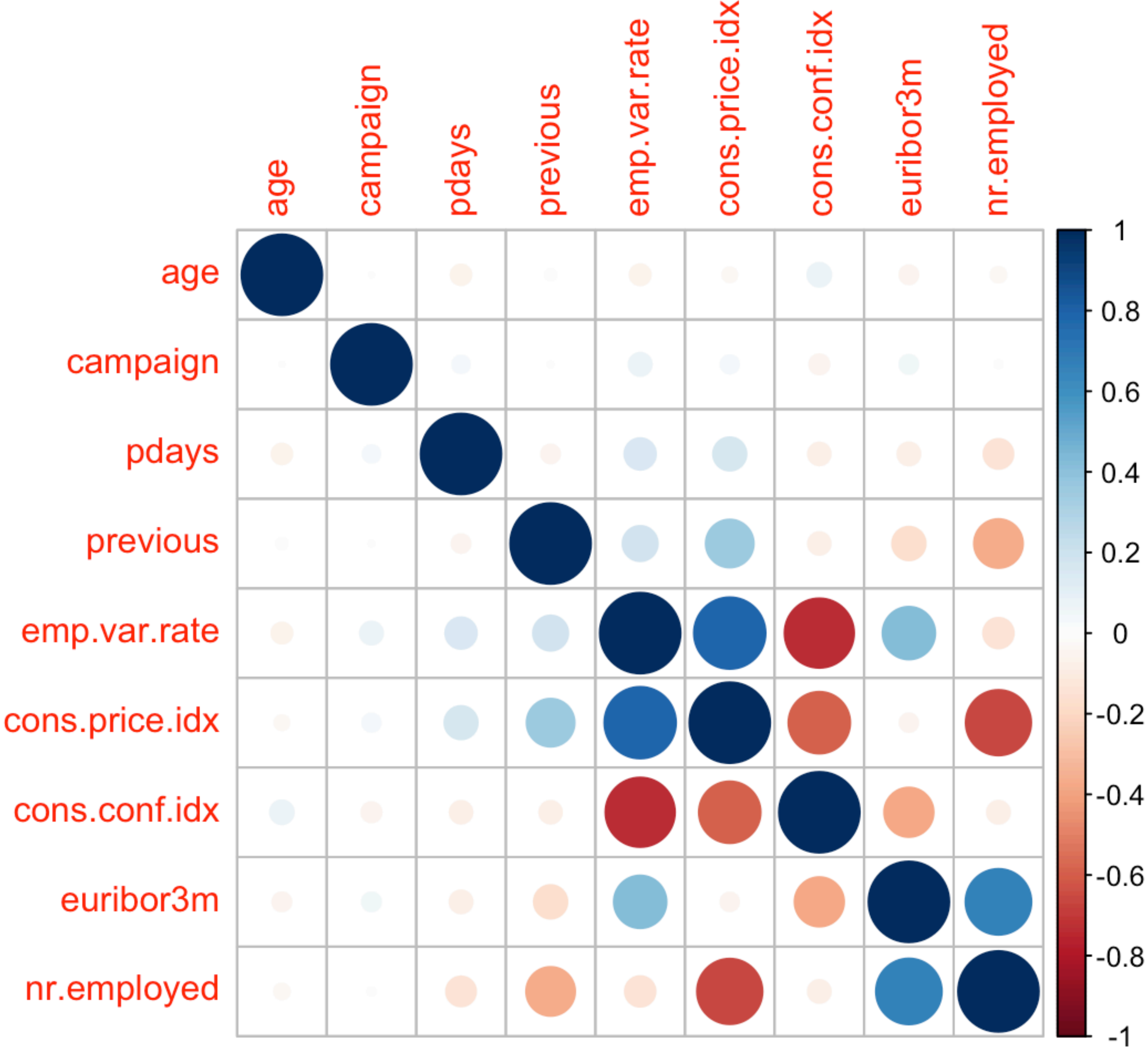




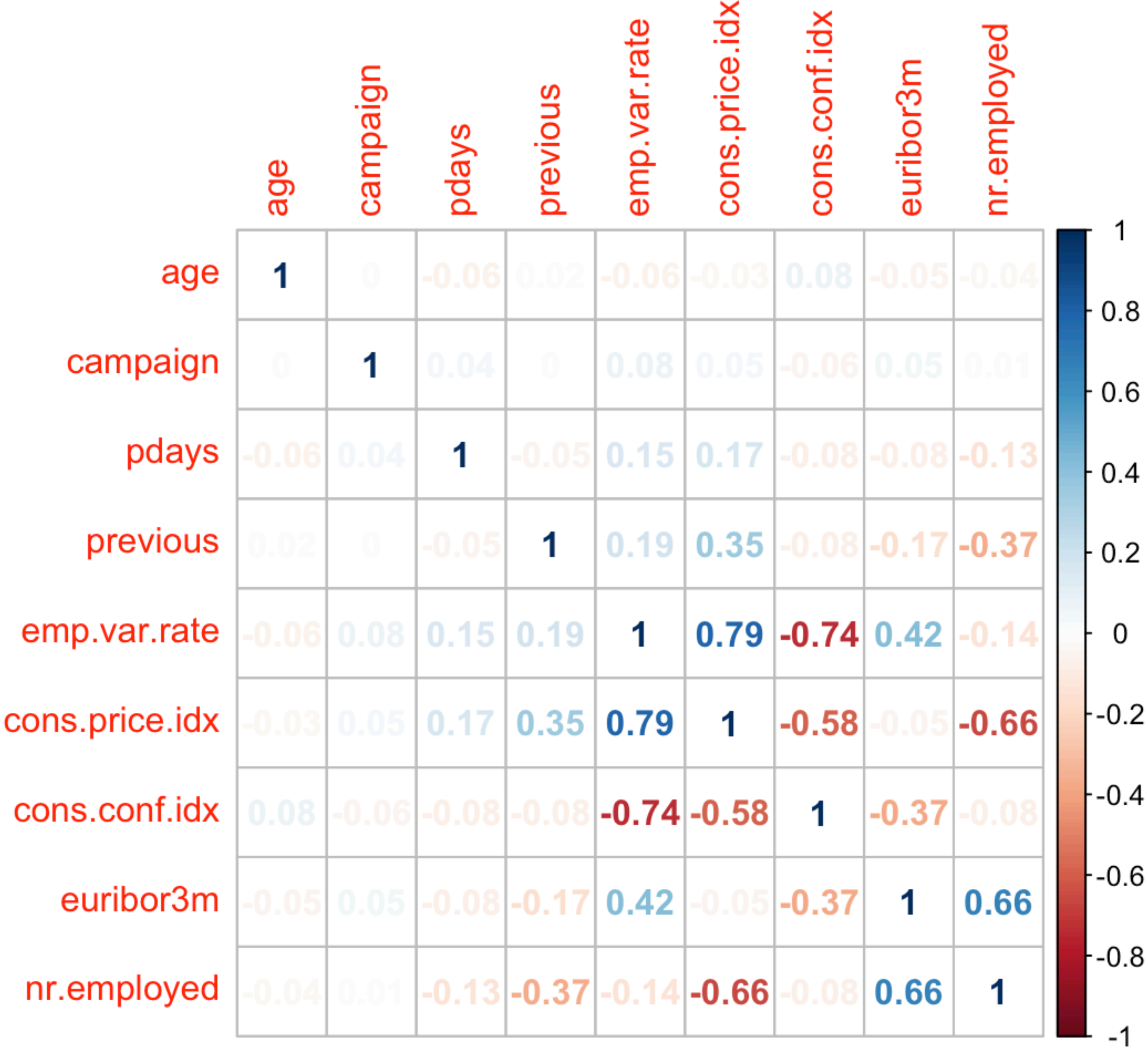
```
numericdata <- subset(bank, select=c("age", "campaign","pdays","previous","emp.var.ra
te","cons.price.idx","cons.conf.idx","euribor3m","nr.employed"))

pairs(numericdata)
```



```
M <- cor(numericdata)
corrplot(M, method = "circle")
```



```
#or view in corr magnitudes
corrplot(M, method = "number")
```



```
#From the correlation plot, we can see that there are good correlations between 'cons
.price.idx'&'emp.var.rate', 'cons.conf.idx'&'emp.var.rate',cons.conf.idx'&'cons.price
.idx','cons.price.idx'&'nr.employed', cons.conf.idx'&'nr.employed','emp.var.rate'& nr
.employed',nr.employed'& euribor3m.
 #Those multicollinearity problems may not affect our predictions but indeed affect c
ausal inferences.
#=============================================
```