Bank_Analysis_and_Clustering.R

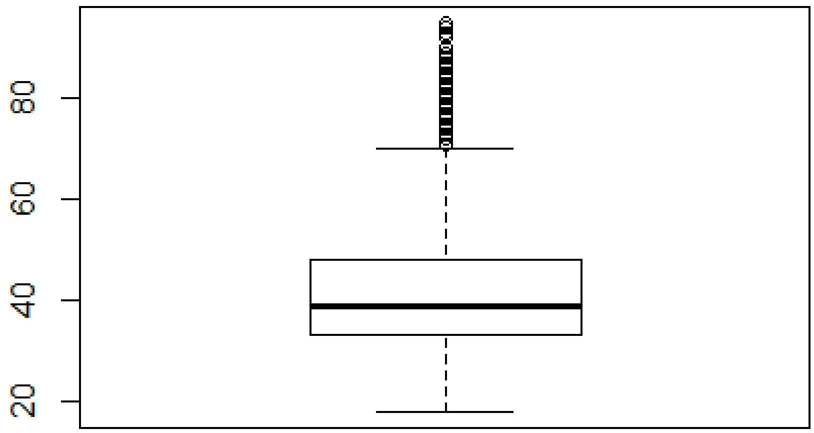
deept

Fri Mar 15 13:59:35 2019

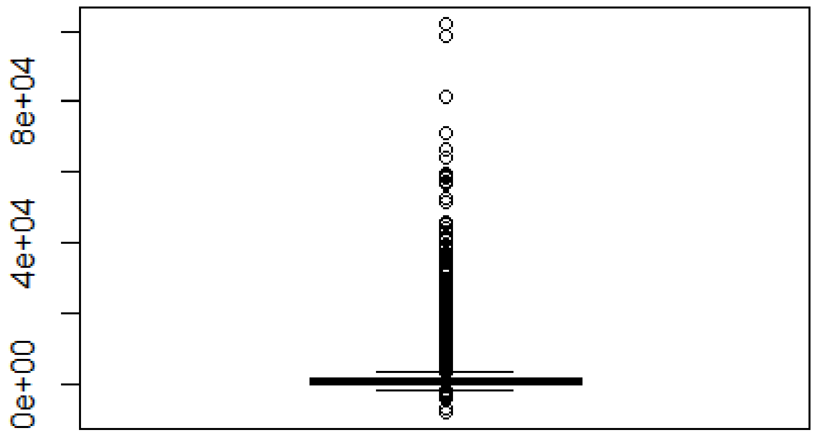
```
bank <- read.csv("~/Spring 19 Sem/Multi Analysis/bank-full.csv", sep=";")
str(bank)
```

```
## '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 10 ...
##  $ 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 ...
```

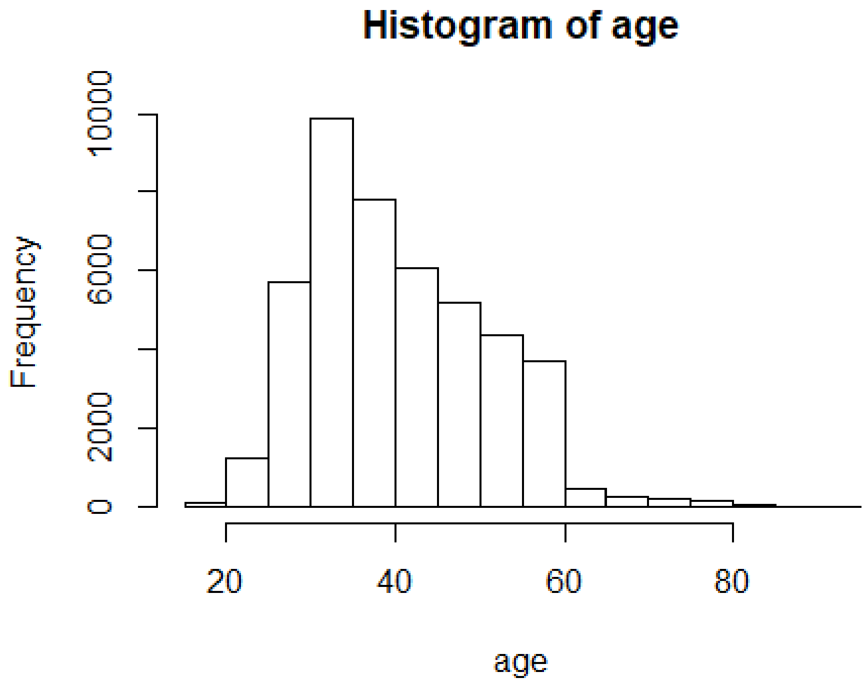
```
attach(bank)
boxplot(age)
```



```
boxplot(balance)
```

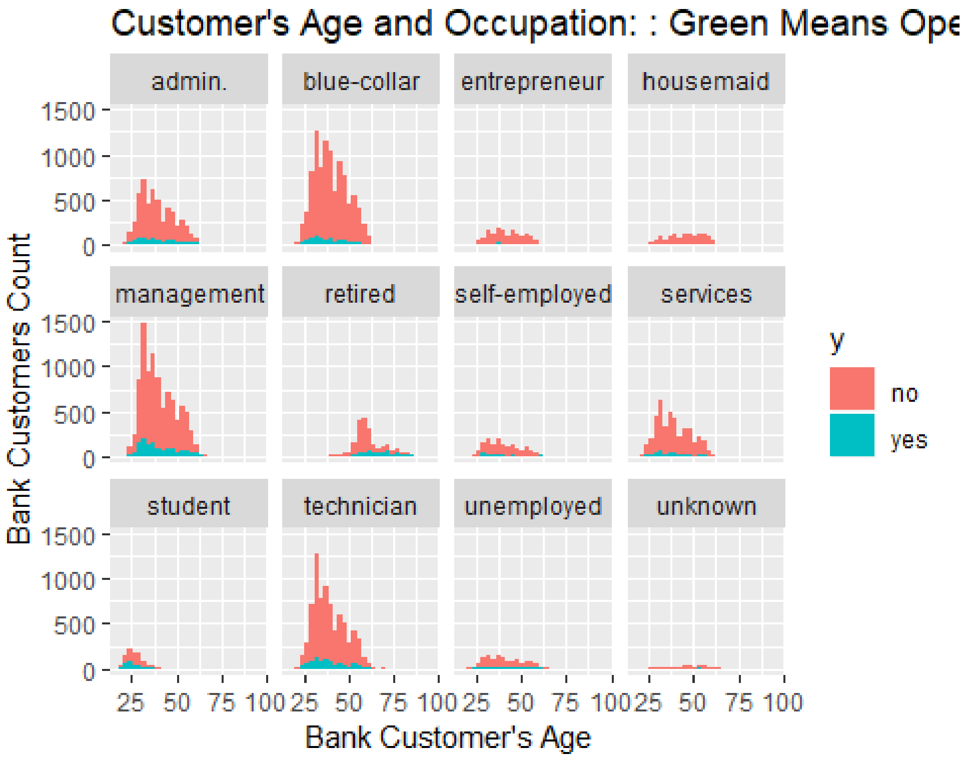


```
hist(age)
library(ggplot2)
```



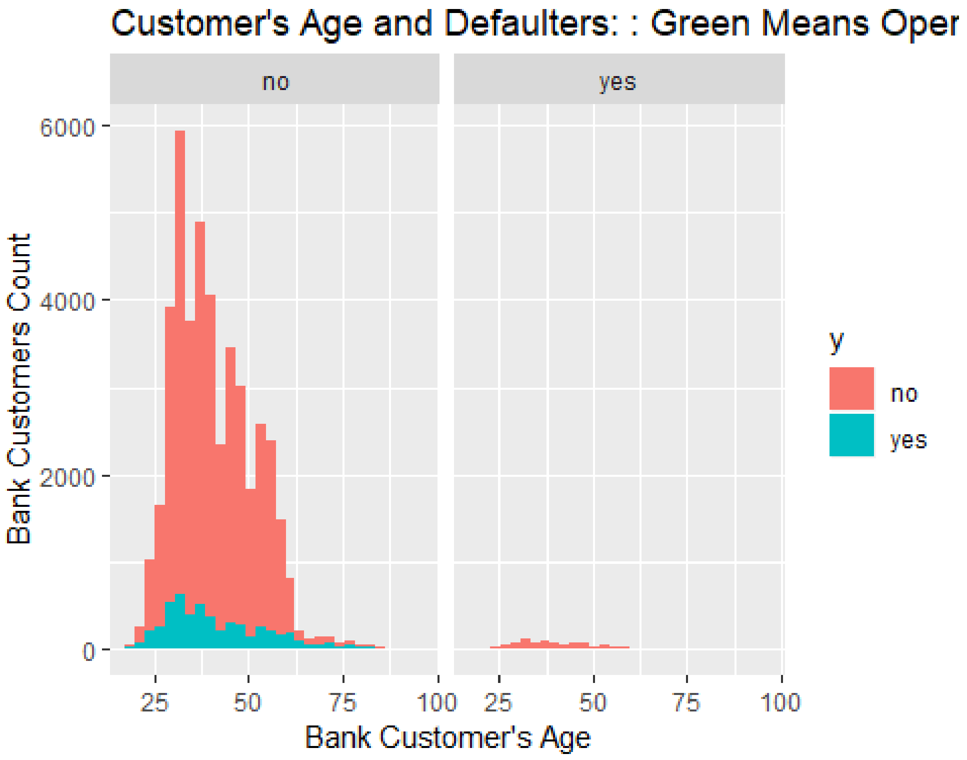
```
ggplot(bank, aes(age, fill=y)) + geom_histogram() + facet_wrap(~job) +
  labs(title = "Customer's Age and Occupation: : Green Means Opened Fixed Dep
osit" ,x="Bank Customer's Age", y="Bank Customers Count")
```

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

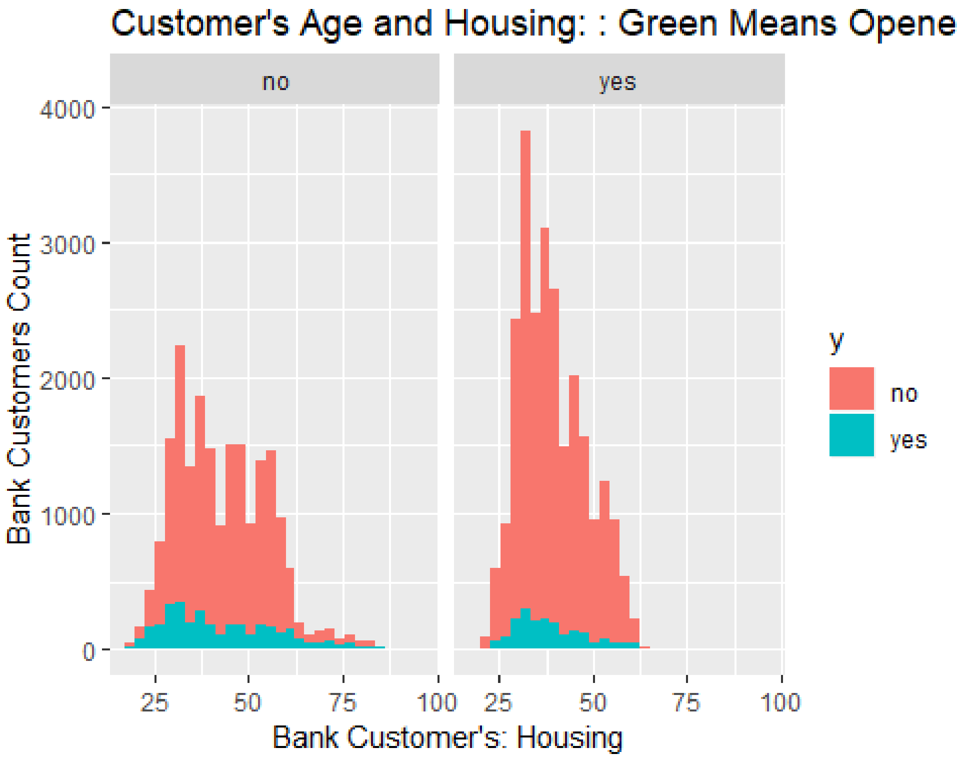


```r
ggplot(bank, aes(age, fill=y)) + geom_histogram() + facet_wrap(~default)+
  labs(title = "Customer's Age and Defaulters: : Green Means Opened Fixed Dep
osit" ,x="Bank Customer's Age", y="Bank Customers Count")
```

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

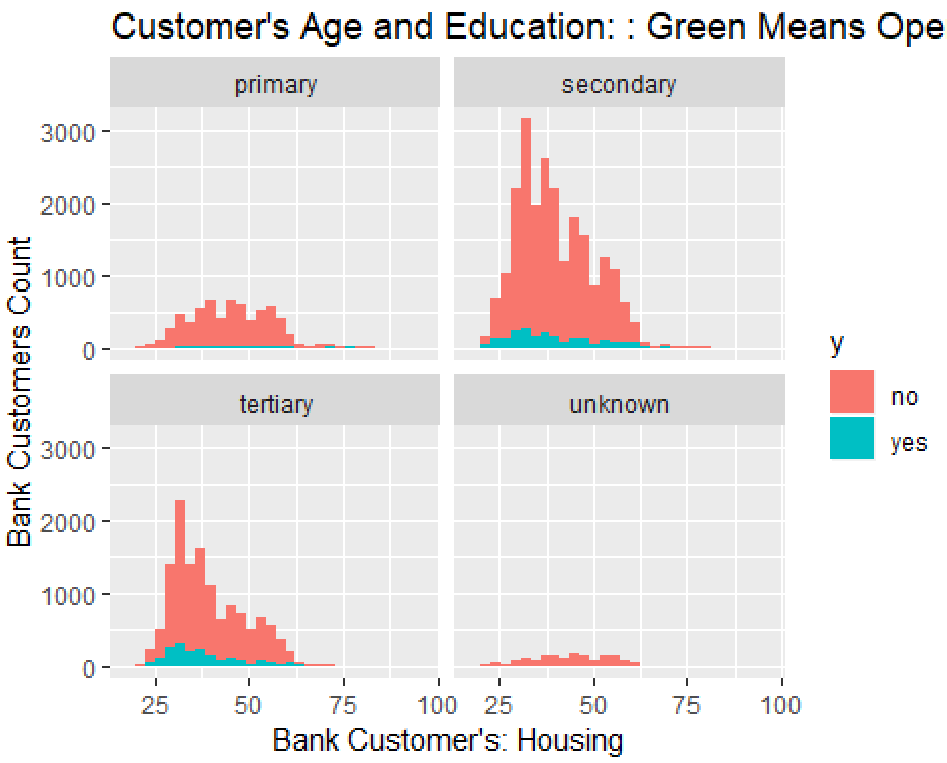


```
ggplot(bank, aes(age, fill=y)) + geom_histogram() + facet_wrap(~housing)+
  labs(title = "Customer's Age and Housing: : Green Means Opened Fixed Deposi
t" ,x="Bank Customer's: Housing", y="Bank Customers Count")

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

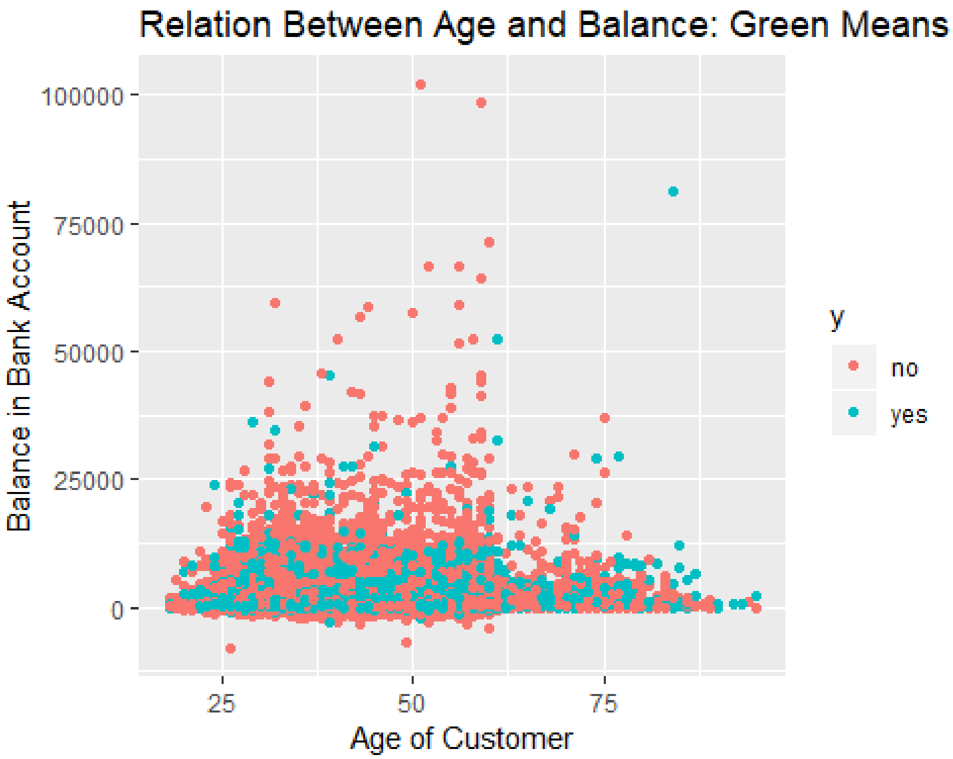


```
ggplot(bank, aes(age, fill=y)) + geom_histogram() + facet_wrap(~education)+
  labs(title = "Customer's Age and Education: : Green Means Opened Fixed Depo
sit" ,x="Bank Customer's: Housing", y="Bank Customers Count")
```

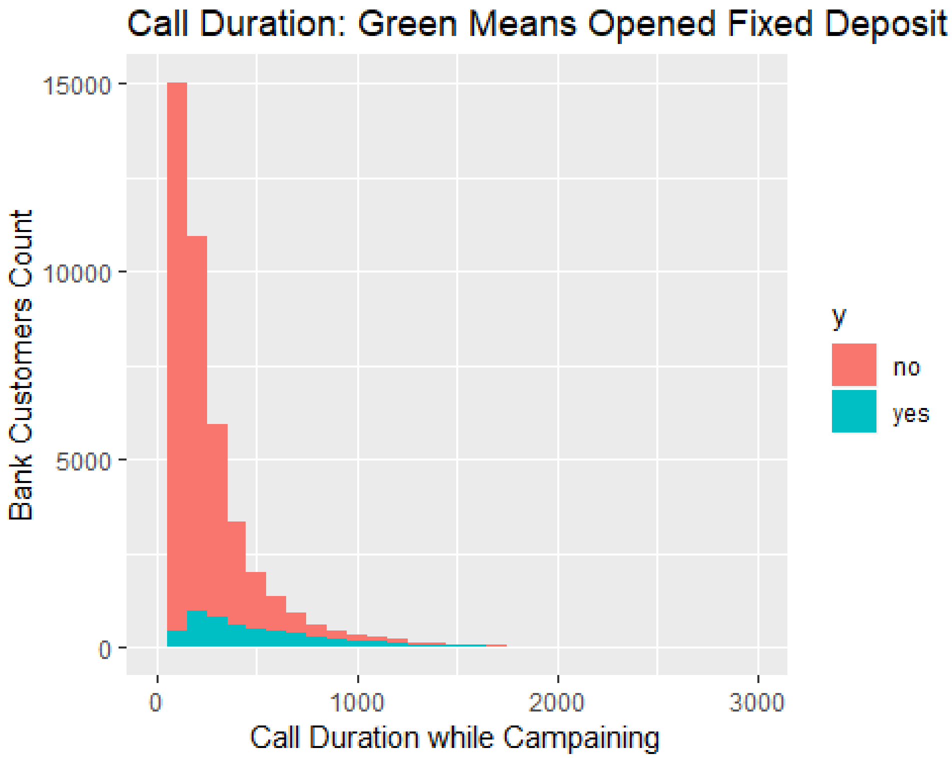
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



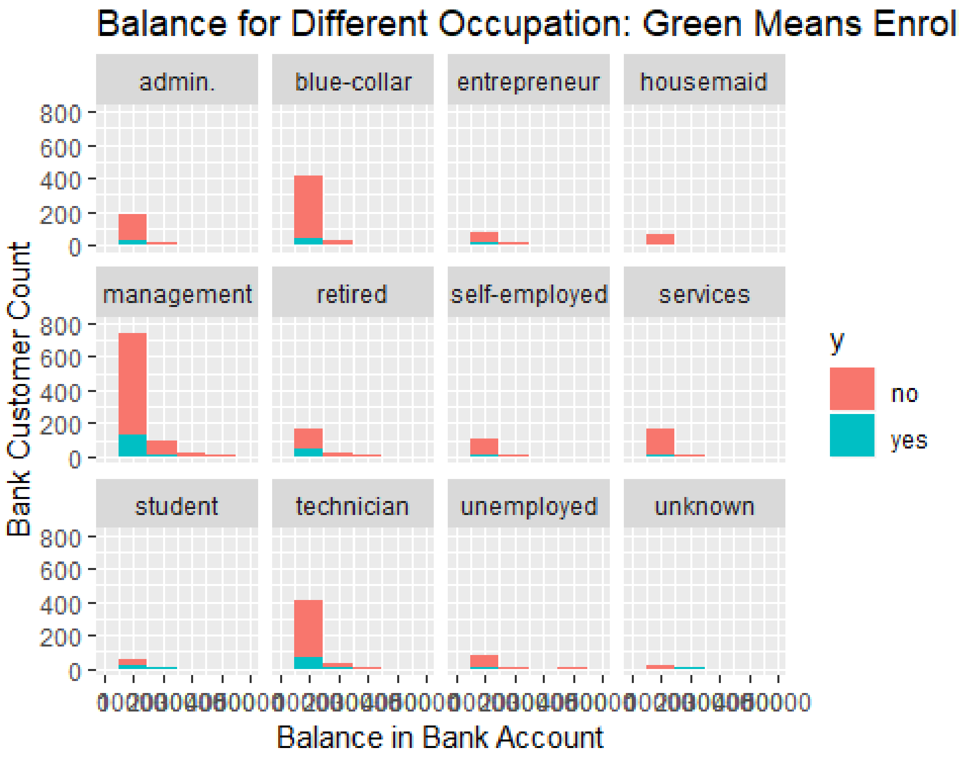
```r
ggplot(bank, aes(age,balance,color=y))  + geom_point() +
  labs(title = "Relation Between Age and Balance: Green Means Opened Fixed De
posit" ,x="Age of Customer", y="Balance in Bank Account")
```



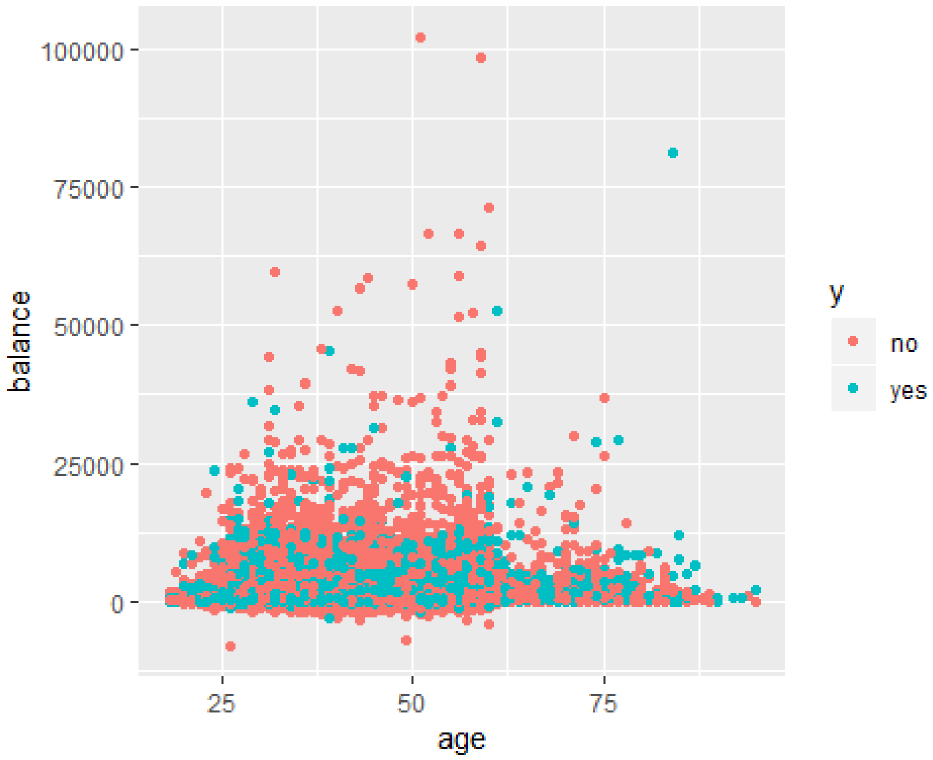
```
ggplot(bank, aes(duration, fill=y)) + geom_histogram(binwidth=100) + xlim(0,3
000)+
labs(title = "Call Duration: Green Means Opened Fixed Deposit" ,x="Call Durat
ion while Campaining", y="Bank Customers Count")

## Warning: Removed 14 rows containing non-finite values (stat_bin).

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



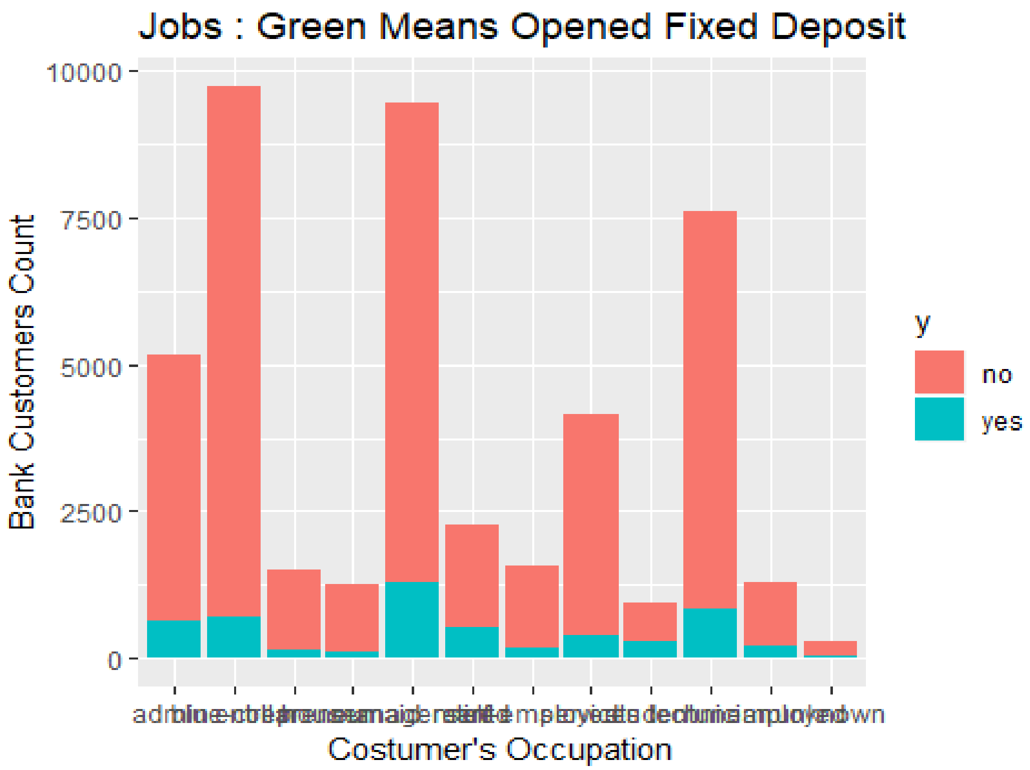
```
ggplot(bank, aes(balance,fill=y)) + geom_histogram(binwidth=10000 ) + xlim(0,
50000) +
  ylim(0,800) + facet_wrap(~job) +
  labs(title = "Balance for Different Occupation: Green Means Enrolled for Fi
xed Deposit" ,x="Balance in Bank Account", y="Bank Customer Count")

## Warning: Removed 3784 rows containing non-finite values (stat_bin).

## Warning: Removed 48 rows containing missing values (geom_bar).
```



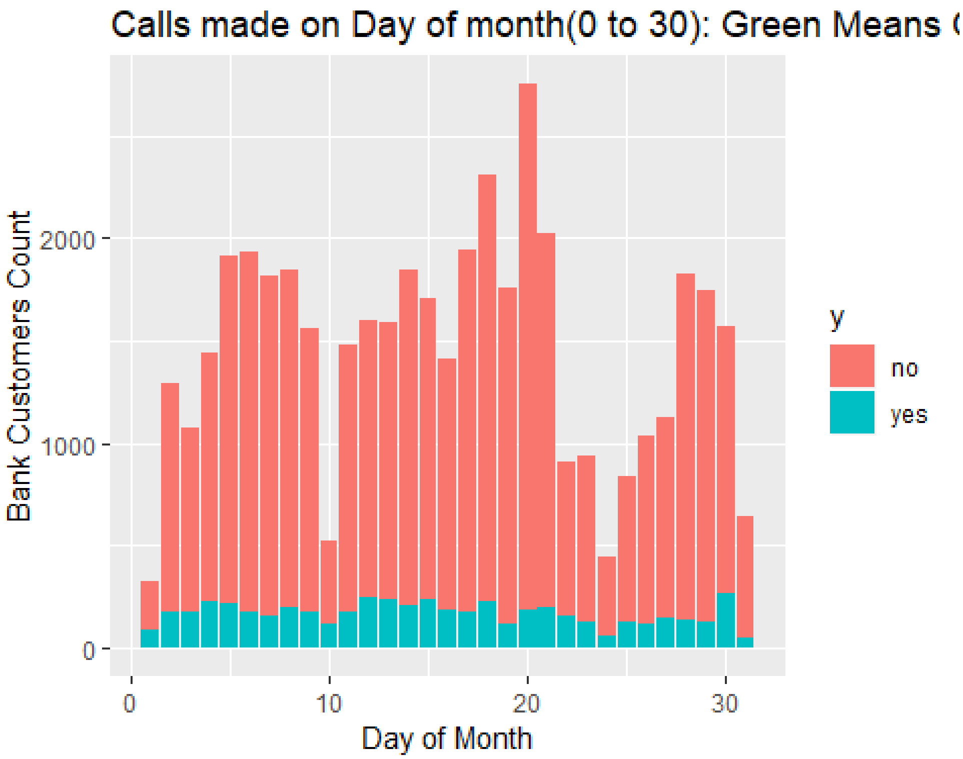
```r
summary(balance)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   -8019      72     448    1362    1428  102127

ggplot(bank, aes(age,balance,color=y))  + geom_point()
```



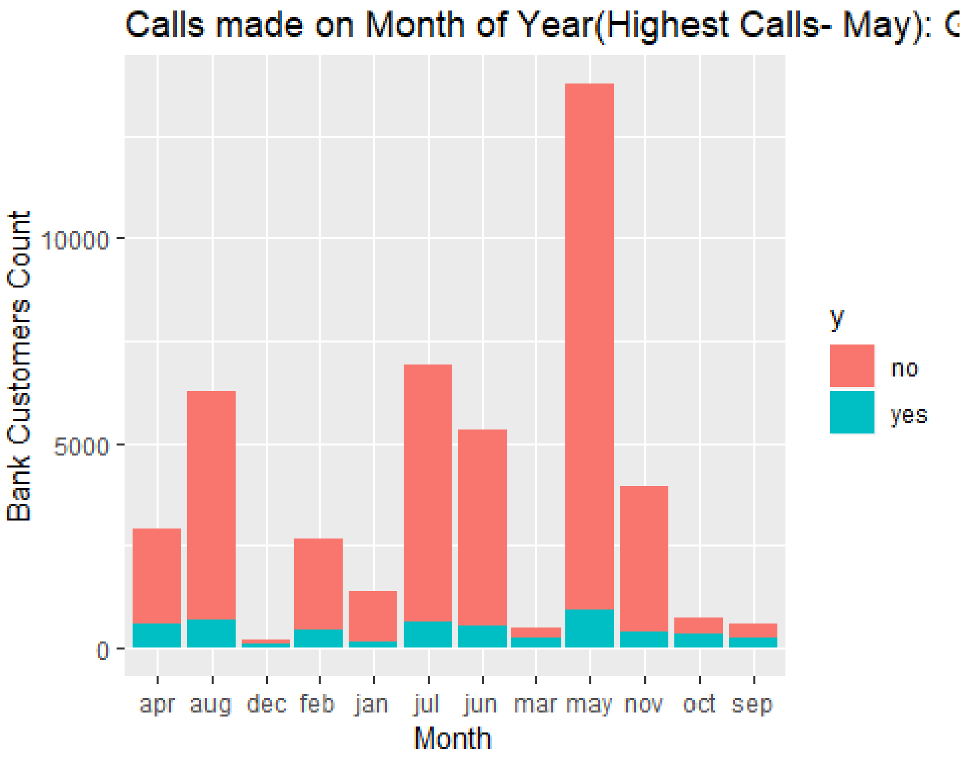
```
ggplot(bank, aes(job, fill=y)) + geom_bar()+
  labs(title = "Jobs : Green Means Opened Fixed Deposit" ,x="Costumer's Occup
ation", y="Bank Customers Count")
```



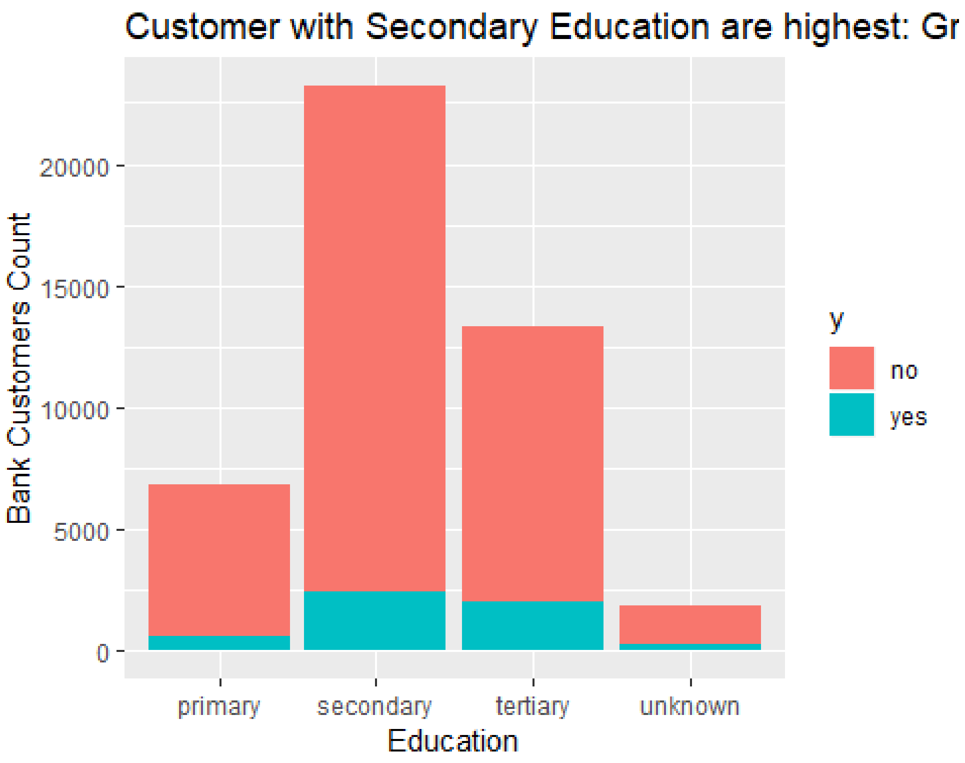
```r
ggplot(bank, aes(day, fill=y)) + geom_bar() +
  labs(title = "Calls made on Day of month(0 to 30): Green Means Opened Fixed
Deposit" ,x="Day of Month", y="Bank Customers Count")
```



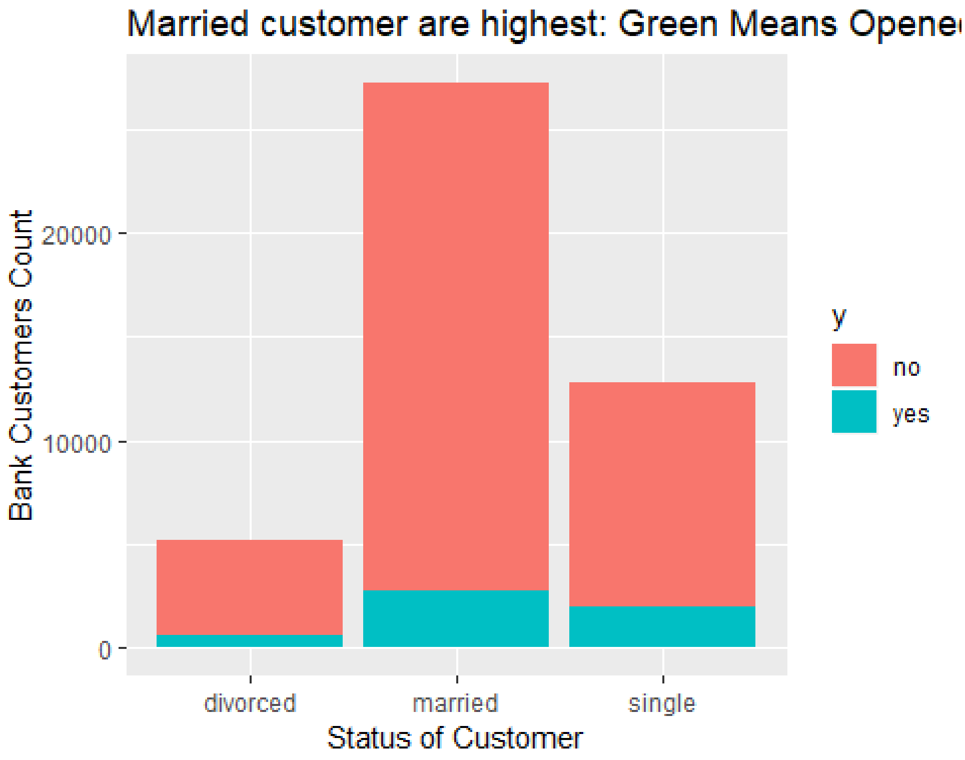
```
ggplot(bank, aes(month, fill=y)) + geom_bar() +
  labs(title = "Calls made on Month of Year(Highest Calls- May): Green Means
Opened Fixed Deposit " ,x="Month", y="Bank Customers Count")
```



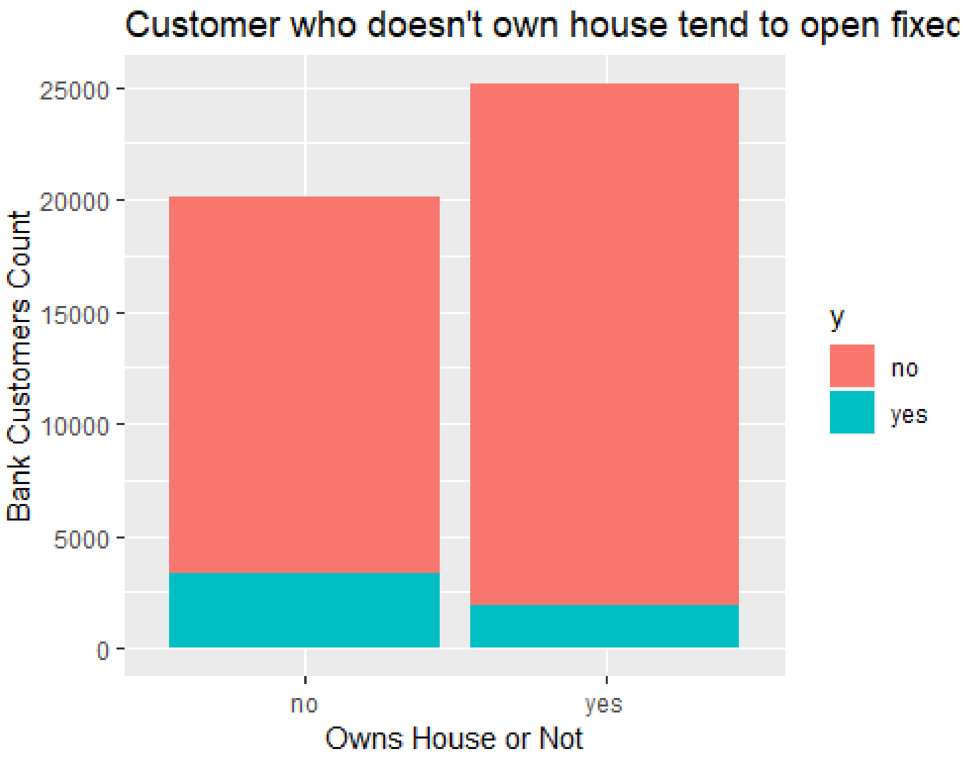
```r
ggplot(bank, aes(education, fill=y)) + geom_bar() +
  labs(title = "Customer with Secondary Education are highest: Green Means Op
ened Fixed Deposit " ,x="Education", y="Bank Customers Count")
```



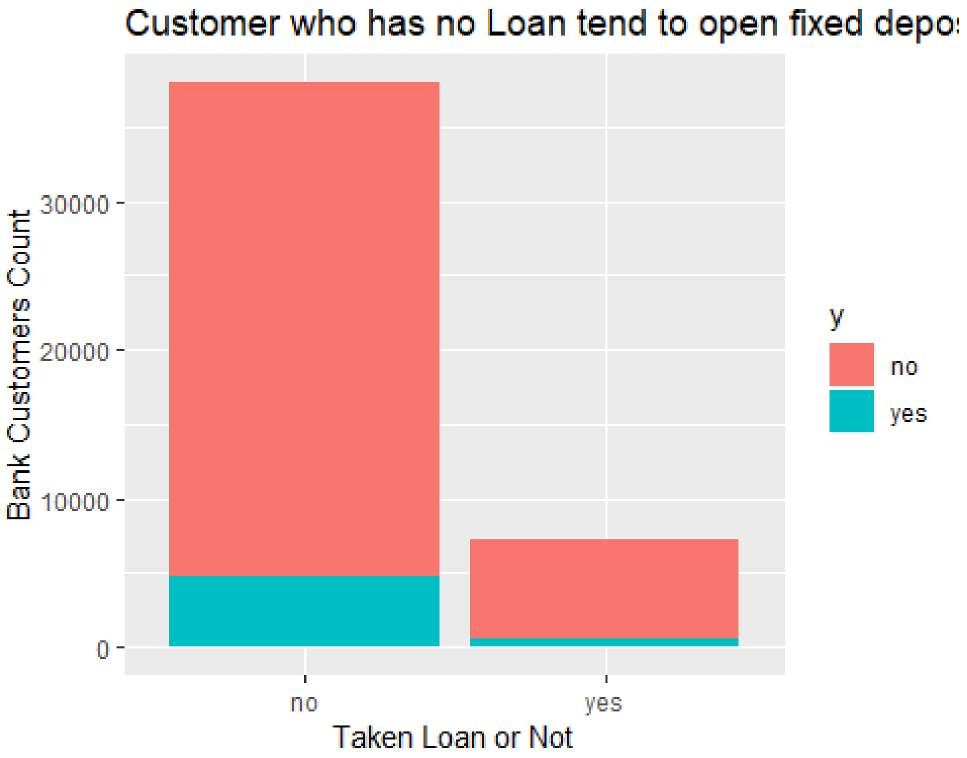
```
ggplot(bank, aes(marital, fill=y)) + geom_bar() +
  labs(title = "Married customer are highest: Green Means Opened Fixed Deposi
t " ,x="Status of Customer", y="Bank Customers Count")
```



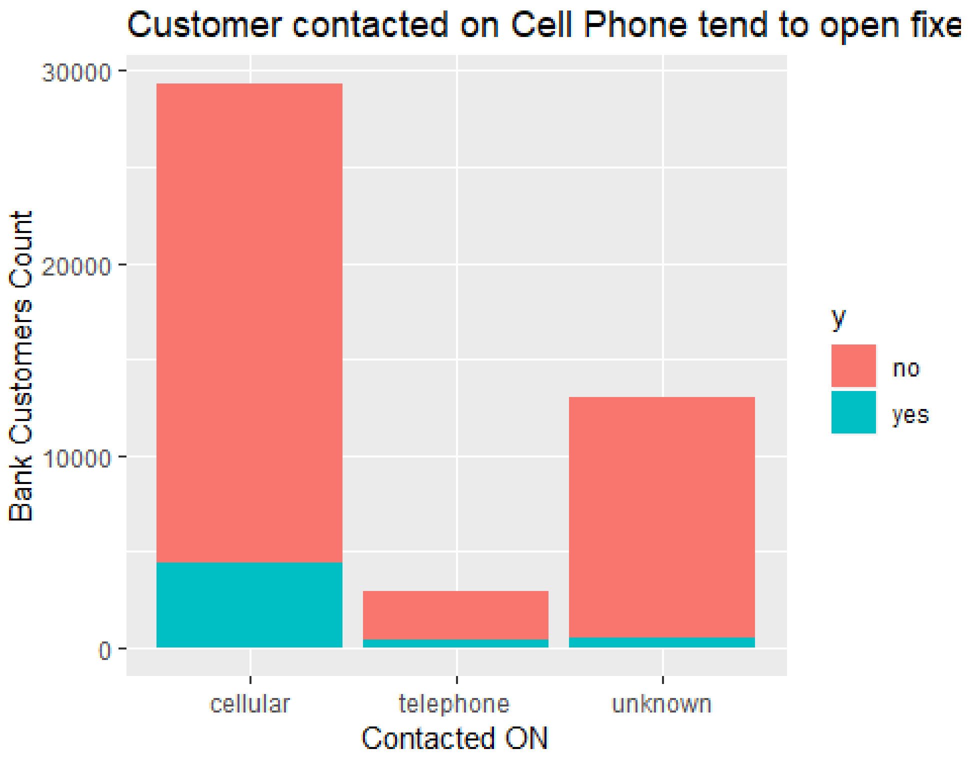
```
ggplot(bank, aes(housing,fill=y)) + geom_bar()+
  labs(title = "Customer who doesn't own house tend to open fixed deposit mor
e" ,x="Owns House or Not", y="Bank Customers Count")
```



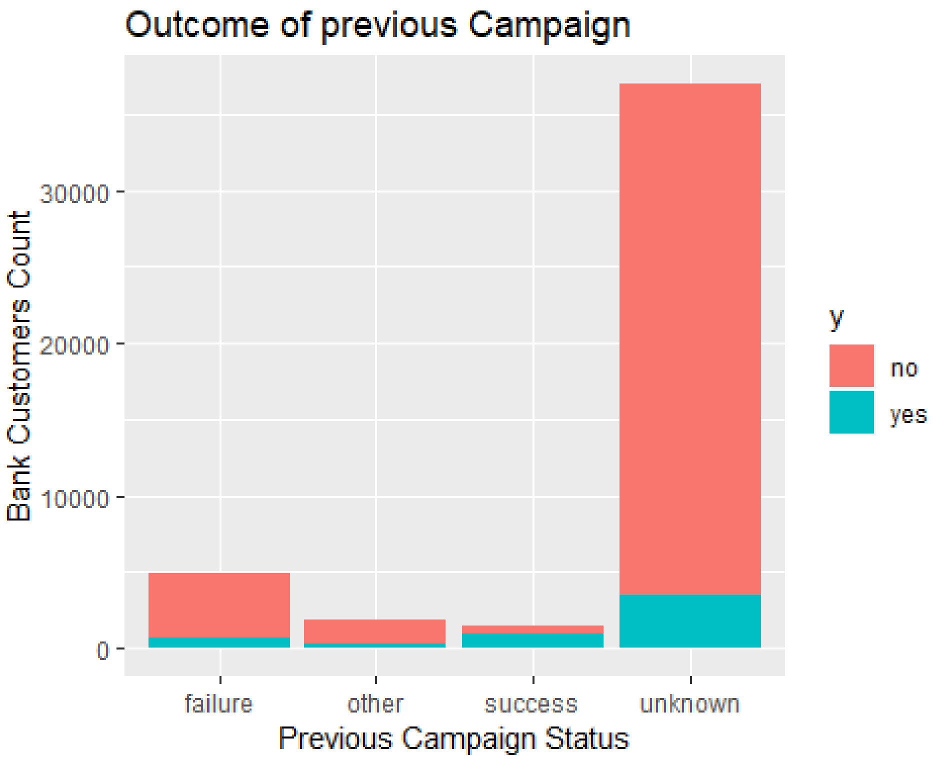
```
ggplot(bank, aes(loan,fill=y)) + geom_bar()+
  labs(title = "Customer who has no Loan tend to open fixed deposit more" ,x=
"Taken Loan or Not", y="Bank Customers Count")
```



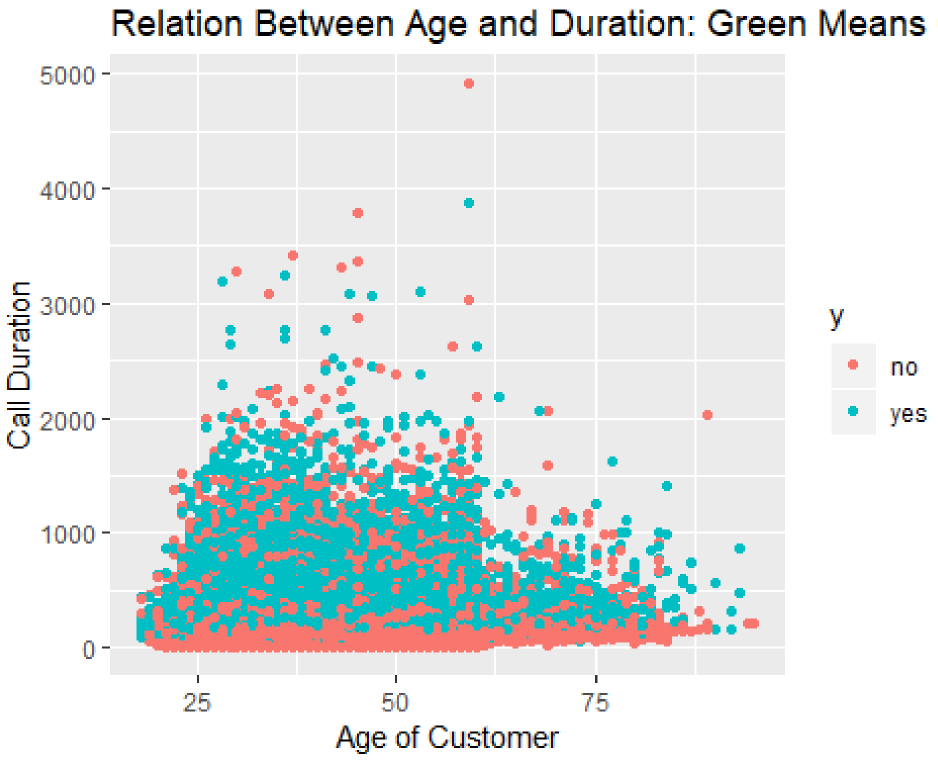
```
ggplot(bank, aes(contact,fill=y)) + geom_bar()+
  labs(title = "Customer contacted on Cell Phone tend to open fixed deposit m
ore" ,x="Contacted ON", y="Bank Customers Count")
```



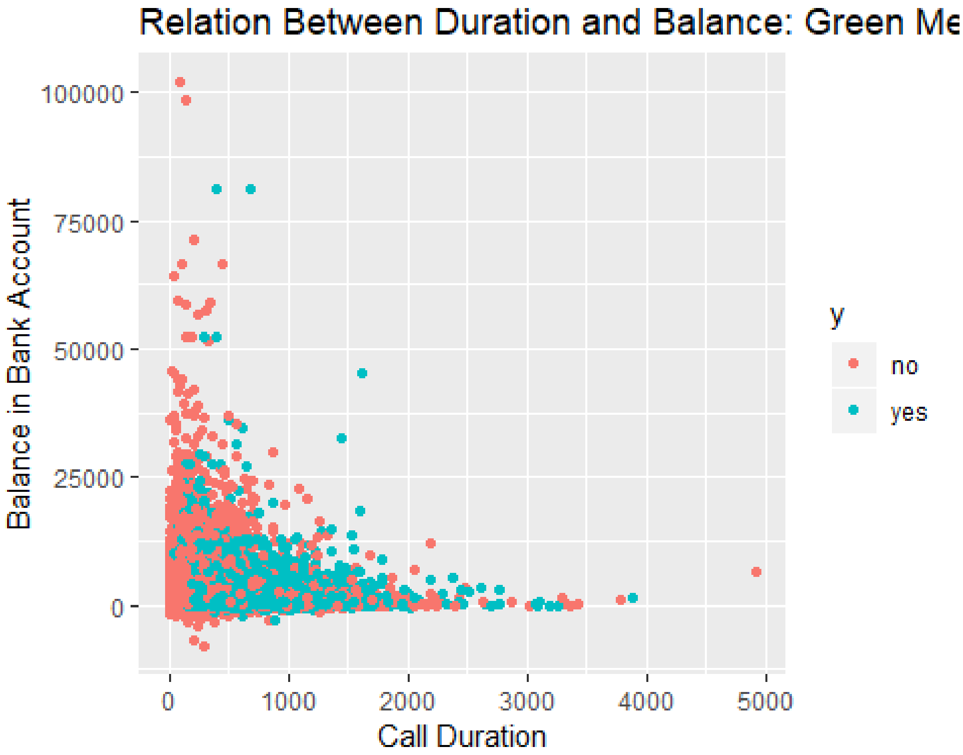
```
ggplot(bank, aes(poutcome,fill=y)) + geom_bar()+
  labs(title = "Outcome of previous Campaign" ,x="Previous Campaign Status",
y="Bank Customers Count")
```



```
ggplot(bank, aes(age,duration,color=y))  + geom_point() +
  labs(title = "Relation Between Age and Duration: Green Means Opened Fixed D
eposit" ,x="Age of Customer", y="Call Duration")
```



```r
ggplot(bank, aes(duration,balance,color=y))  + geom_point() +
  labs(title = "Relation Between Duration and Balance: Green Means Opened Fix
ed Deposit" ,x="Call Duration", y="Balance in Bank Account")
```



```
#Converting Factors to Numeric
bank_modified=bank

#unknown = 0 ,student=1, unemployed=2, housemaid=3
#self-employed 4, entrepreneur 5, retired 6, management 7,services 8 , techni
cian 9
# admin 11   blue-collar    10
bank_job= ifelse(bank$job== 'admin.', 11,
                 ifelse(bank$job=='blue-collar', 10,
                        ifelse(bank$job=='technician',9,
                               ifelse(bank$job=='services',8,
                                      ifelse(bank$job=='management',7,
                                             ifelse(bank$job=='retired',6,
                                                    ifelse(bank$job=='entrepr
eneur',5,
                                                           ifelse(bank$job=='
self-employed',4,
                                                                  ifelse(bank
$job=='housemaid',3,
                                                                         ifel
se(bank$job=='unemployed',2,

ifelse(bank$job=='student',1,0)))))))))))

#added column in new dataframe bank_modified
bank_modified=cbind(bank_modified,bank_job)
#head(bank_modified[,c('education','bank_education')],30)

#month from factor to numeric
unique(bank$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
```

```
#may jun jul aug oct nov dec mar apr sep
bank_month=ifelse(bank$month=='mar',3,
                  ifelse(bank$month=='apr',4,
                         ifelse(bank$month=='may',5,
                                ifelse(bank$month=='jun',6,
                                       ifelse(bank$month=='jul',7,
                                              ifelse(bank$month=='aug',8,
                                                     ifelse(bank$month=='sep'
,9,
                                                            ifelse(bank$month
=='oct',10,
                                                                   ifelse(ban
k$month=='nov',11,
                                                                          ife
lse(bank$month=='dec',12,0))))))))))
```

```
#adding it to data frame bank_modified
bank_modified=cbind(bank_modified,bank_month)




#loan from factor to numric

bank_loan= ifelse(bank$loan=='yes',1,0)
bank_modified=cbind(bank_modified,bank_loan)

#default from factor to numric
bank_default= ifelse(bank$default=='yes',1,0)
bank_modified=cbind(bank_modified,bank_default)

unique(bank_modified$education)

## [1] tertiary  secondary unknown   primary
## Levels: primary secondary tertiary unknown

#education from factor to numeric in the order of highest count: higher count
get the highest number
bank_education=ifelse(bank$education=='secondary',1,
                      ifelse(bank$education=='tertiary',2,
                             ifelse(bank$education=='primary',3,0)))

bank_modified=cbind(bank_modified,bank_education)


bank_contact=ifelse(bank$contact=='cellular',2,1)
bank_modified=cbind(bank_modified,bank_contact)

#changing marital from factor to integer
#married 3, single 2, divorced 1 and unknown 0
bank_marital=ifelse(bank$marital=='married',3,
                    ifelse(bank$marital=='single',2,
                           ifelse(bank$marital=='divorced',1,0)))

bank_modified=cbind(bank_modified,bank_marital)


#Housing from factor to numeric
bank_housing= ifelse(bank$housing=='yes',1,0)

bank_modified=cbind(bank_modified,bank_housing)
head(bank_modified)
```

```
##   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 bank_job
## 1   5   may      261        1    -1        0  unknown no        7
## 2   5   may      151        1    -1        0  unknown no        9
## 3   5   may       76        1    -1        0  unknown no        5
## 4   5   may       92        1    -1        0  unknown no       10
## 5   5   may      198        1    -1        0  unknown no        0
## 6   5   may      139        1    -1        0  unknown no        7
##   bank_month bank_loan bank_default bank_education bank_contact
## 1          5         0            0              2            1
## 2          5         0            0              1            1
## 3          5         1            0              1            1
## 4          5         0            0              0            1
## 5          5         0            0              0            1
## 6          5         0            0              2            1
##   bank_marital bank_housing
## 1            3            1
## 2            2            1
## 3            3            1
## 4            3            1
## 5            2            0
## 6            3            1
```

```
bank_y=ifelse(bank$y=='yes',1,0)
bank_modified=cbind(bank_modified,bank_y)


bank_int = bank_modified[,c('y','age','duration','campaign','balance','pdays'
,'previous','bank_housing','bank_loan','bank_job','bank_education','bank_mont
h','bank_contact','bank_marital','bank_default','day')]


str(bank_int)
```

```
## 'data.frame':    45211 obs. of  16 variables:
##  $ y             : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ age           : int  58 44 33 47 33 35 28 42 58 43 ...
##  $ duration      : int  261 151 76 92 198 139 217 380 50 55 ...
##  $ campaign      : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ balance       : int  2143 29 2 1506 1 231 447 2 121 593 ...
##  $ pdays         : int  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
##  $ previous      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ bank_housing  : num  1 1 1 1 0 1 1 1 1 1 ...
##  $ bank_loan     : num  0 0 1 0 0 0 1 0 0 0 ...
##  $ bank_job      : num  7 9 5 10 0 7 7 5 6 9 ...
##  $ bank_education: num  2 1 1 0 0 2 2 2 3 1 ...
##  $ bank_month    : num  5 5 5 5 5 5 5 5 5 5 ...
##  $ bank_contact  : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ bank_marital  : num  3 2 3 3 2 3 2 1 3 2 ...
##  $ bank_default  : num  0 0 0 0 0 0 0 1 0 0 ...
##  $ day           : int  5 5 5 5 5 5 5 5 5 5 ...
```

```
#Clustering

# Standardizing the data with scale()

bank_int_scale <- scale(bank_int[-1])
# K-means, k=2, 3, 4, 5, 6
# Centers (k's) are numbers thus, 10 random sets are chosen

(kmeans2 <- kmeans(bank_int_scale,2,nstart = 10))
```

```
## K-means clustering with 2 clusters of sizes 6779, 38432
## 
## Cluster means:
##            age     duration    campaign      balance      pdays   previous
## 1 -0.040697170  0.009188141 -0.21370280  0.046276409  2.1273716  1.2552644
## 2  0.007178552 -0.001620691  0.03769492 -0.008162671 -0.3752459 -0.2214154
##   bank_housing    bank_loan     bank_job bank_education  bank_month
## 1   0.28767123 -0.055566372  0.049181875   -0.045352216 -0.31322588
## 2  -0.05074217  0.009801323 -0.008675165    0.007999653  0.05524975
##   bank_contact bank_marital bank_default         day
## 1    0.5910606  -0.04507934  -0.08005225 -0.23109110
## 2   -0.1042569   0.00795152   0.01412037  0.04076204
## 
## Clustering vector:
##     [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2
##    [35] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2
##    [69] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2
2
## [44983] 1 1 1 2 2 1 2 1 1 1 2 1 2 2 1 1 1 1 2 2 1 1 1 1 1 1 2 2 2 1 1 1 1
1
## [45017] 2 2 2 1 2 1 1 1 1 1 1 2 1 2 1 1 1 2 1 2 1 1 2 1 2 2 2 2 2 2 2 2 2
2
## [45051] 1 2 2 2 1 2 1 1 2 2 1 2 2 1 1 2 1 1 1 1 2 1 2 1 2 1 2 1 1 2 2 1 2
2
## [45085] 1 2 2 2 2 2 2 2 2 1 1 1 1 1 1 2 2 1 1 1 2 1 1 1 2 1 1 2 2 2 1 2 2
1
## [45119] 2 2 1 1 2 1 2 2 1 2 2 2 2 2 1 2 1 2 1 2 1 2 2 1 1 2 2 2 1 1 1 1 1
1
## [45153] 1 2 1 2 2 1 1 1 1 2 2 2 2 1 1 2 1 1 2 1 1 1 2 1 2 1 1 1 2 2 1 1 2
```

```
1
## [45187] 2 2 1 1 1 1 2 2 1 1 2 2 2 1 2 1 2 2 1 2 2 2 1 2 1
##
## Within cluster sum of squares by cluster:
## [1] 113181.7 511134.2
##  (between_SS / total_SS =   7.9 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
# Computing the percentage of variation accounted for. Two clusters
perc.var.2 <- round(100*(1 - kmeans2$betweenss/kmeans2$totss),1)
names(perc.var.2) <- "Perc. 2 clus"
perc.var.2
```

```
## Perc. 2 clus
##         92.1
```

```
# Computing the percentage of variation accounted for. Three clusters
(kmeans3 <- kmeans(bank_int_scale,3,nstart = 10))
```

```
## K-means clustering with 3 clusters of sizes 20429, 5958, 18824
##
## Cluster means:
##          age    duration    campaign     balance      pdays   previous
## 1 -0.1753871  0.01590494  0.01956880 -0.08315380 -0.3870016 -0.2312393
## 2 -0.1145935 -0.01174023 -0.20798358 -0.00640250  2.2756391  1.2952172
## 3  0.2266113 -0.01354514  0.04459175  0.09227024 -0.3002657 -0.1589947
##   bank_housing   bank_loan   bank_job bank_education   bank_month
## 1    0.8921322  0.04573574  0.1911619    -0.02945806 -0.07032286
## 2    0.4378858 -0.02636883  0.1300516    -0.05463458 -0.43764141
## 3   -1.1067941 -0.04128931 -0.2486238     0.04926219  0.21483708
##   bank_contact bank_marital bank_default         day
## 1   -0.3160002   0.02337750  0.009468112  0.02878882
## 2    0.5979454  -0.04629391 -0.074936274 -0.25833543
## 3    0.1536872  -0.01071823  0.013442747  0.05052251
##
## Clustering vector:
##     [1] 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 1 1
1
2
## [45119] 3 3 2 2 1 3 1 3 2 3 3 3 3 3 3 3 3 3 2 3 2 3 3 3 2 3 3 3 2 2 2 2 2
2
## [45153] 2 3 2 3 3 3 3 3 2 1 3 3 3 2 2 3 2 2 3 2 2 2 3 3 1 2 1 3 3 3 2 3 3
2
## [45187] 3 1 2 3 2 2 3 3 2 2 3 1 3 2 1 3 3 3 3 3 3 3 3 3 2
##
## Within cluster sum of squares by cluster:
```

```
## [1] 228440.29  96955.11 253407.95
##  (between_SS / total_SS =  14.6 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
perc.var.3 <- round(100*(1 - kmeans3$betweenss/kmeans3$totss),1)
names(perc.var.3) <- "Perc. 3 clus"
perc.var.3
```

```
## Perc. 3 clus
##         85.4
```

```
# Computing the percentage of variation accounted for. Four clusters
(kmeans4 <- kmeans(bank_int_scale,4,nstart = 10))
```

```
## Warning: Quick-TRANSfer stage steps exceeded maximum (= 2260550)
```

```
## K-means clustering with 4 clusters of sizes 17086, 6941, 20369, 815
##
## Cluster means:
##          age      duration    campaign     balance       pdays    previous
## 1  0.2313808  0.0005914881  0.01639850  0.15850724 -0.13269238 -0.03695455
## 2 -0.0334067 -0.0254332436  0.01581994 -0.18309538 -0.04336604 -0.02070432
## 3 -0.1774240  0.0111299767 -0.02411385 -0.05085712  0.13493622  0.04346645
## 4 -0.1319590 -0.0739638262  0.12415204 -0.49261476 -0.22126396 -0.13528096
##   bank_housing  bank_loan     bank_job bank_education  bank_month
## 1  -1.10028640 -0.4364795 -0.303559145     0.05513169  0.15697569
## 2   0.10900140  2.2893338  0.174698904    -0.05917204  0.04318659
## 3   0.88758198 -0.4367986  0.194995036    -0.02402602 -0.15092203
## 4  -0.04446938  0.5700306  0.002665732    -0.05138757  0.11323447
##   bank_contact bank_marital bank_default         day
## 1   0.17750176 -0.001082521   -0.1354884  0.03343601
## 2   0.03047211  0.051225989   -0.1354884  0.02282727
## 3  -0.15630533 -0.011113024   -0.1354884 -0.03860857
## 4  -0.07425617 -0.135831222    7.3805430  0.06955349
##
## Clustering vector:
##     [1] 3 3 2 3 1 3 2 4 3 3 3 3 3 3 3 3 3 3 3 3 2 3 2 3 2 3 3 2 3 2 2 3 2
3
##    [35] 3 3 3 1 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 2 2 3 3 3 3 3 3 3 3 3 3 1
2
## [45153] 1 2 1 1 1 1 1 1 3 3 1 1 1 3 1 1 3 3 1 3 1 1 1 1 3 3 3 1 1 1 1 1 1
3
## [45187] 1 3 3 1 1 1 1 1 2 1 1 3 1 3 3 1 1 1 1 2 1 1 1 1 1
##
## Within cluster sum of squares by cluster:
## [1] 215084.11  80004.78 238882.81  10332.42
##  (between_SS / total_SS =  19.7 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
perc.var.4 <- round(100*(1 - kmeans4$betweenss/kmeans4$totss),1)
names(perc.var.4) <- "Perc. 4 clus"
perc.var.4
```

```
## Perc. 4 clus
##         80.3
```

```
# Computing the percentage of variation accounted for. Five clusters
(kmeans5 <- kmeans(bank_int_scale,5,nstart = 10))
```

```
## Warning: Quick-TRANSfer stage steps exceeded maximum (= 2260550)

## Warning: Quick-TRANSfer stage steps exceeded maximum (= 2260550)

## Warning: Quick-TRANSfer stage steps exceeded maximum (= 2260550)
```

```
## K-means clustering with 5 clusters of sizes 5346, 14614, 13496, 10940, 815
##
## Cluster means:
##           age     duration    campaign     balance      pdays   previous
## 1 -0.10778943 -0.008387781 -0.19922040 -0.03089136  2.3859432  1.3448374
## 2  0.03575288 -0.029950553  0.03315848 -0.06032732 -0.3950416 -0.2386587
## 3  0.35088704 -0.065521990  0.19567167  0.22468886 -0.2111507 -0.1009110
## 4 -0.41812387  0.130448241 -0.19757901 -0.14480358 -0.3612518 -0.2038020
## 5 -0.13195897 -0.073963826  0.12415204 -0.49261476 -0.2212640 -0.1352810
##   bank_housing   bank_loan     bank_job bank_education bank_month
## 1   0.41316322 -0.04261600  0.143266954   -0.053033100 -0.6212811
## 2   0.26564962 -0.01595871  0.115600230   -0.005783073 -0.1898741
## 3  -0.64072173 -0.05381501 -0.274037131    0.123920421  0.9140295
## 4   0.23696973  0.06606564  0.113431962   -0.115404053 -0.5787788
## 5  -0.04446938  0.57003064  0.002665732   -0.051387574  0.1132345
##   bank_contact bank_marital bank_default         day
## 1   0.58393446  -0.04379266   -0.1354884 -0.30082099
## 2  -1.35601488   0.03243480   -0.1354884 -0.03147391
## 3   0.64405853   0.11758504   -0.1354884  0.32632235
## 4   0.73705601  -0.15686580   -0.1354884 -0.21870053
## 5  -0.07425617  -0.13583122    7.3805430  0.06955349
##
## Clustering vector:
##     [1] 2 2 2 2 2 2 2 5 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2
##    [35] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2
```

```
1
## [45119] 3 3 1 1 2 3 4 3 1 3 3 3 3 3 3 3 3 3 1 3 1 3 3 3 1 3 3 3 1 1 3 3 1
1
## [45153] 1 3 1 3 3 3 3 3 1 3 3 3 3 1 1 3 3 1 3 1 3 3 3 3 3 1 3 3 3 3 1 3 3
3
## [45187] 3 3 1 3 3 3 3 3 1 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 1
##
## Within cluster sum of squares by cluster:
## [1]  85440.61 146561.66 165289.49 102106.20  10332.42
##  (between_SS / total_SS =  24.8 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
perc.var.5 <- round(100*(1 - kmeans5$betweenss/kmeans5$totss),1)
names(perc.var.5) <- "Perc. 5 clus"
perc.var.5
```

```
## Perc. 5 clus
##         75.2
```

```
(kmeans6 <- kmeans(bank_int_scale,6,nstart = 10))
```

```
## K-means clustering with 6 clusters of sizes 1356, 16036, 5404, 6366, 1019,
15030
##
## Cluster means:
##           age     duration    campaign     balance       pdays    previous
## 1 -0.04635733 -0.433358515  4.23317932 -0.14160621 -0.40121100 -0.24009177
## 2 -0.18409451  0.036158383 -0.14164590 -0.12950254 -0.37728575 -0.22521712
## 3 -0.13073866 -0.008998678 -0.20152039 -0.08119601  2.37537248  1.33613479
## 4 -0.02696657 -0.012473399 -0.10605464 -0.22100209 -0.30550108 -0.17924109
## 5  0.33445734  0.039245441 -0.08722325  4.68451880 -0.08450932 -0.01633837
## 6  0.23635189  0.006376695 -0.10749947 -0.04385371 -0.28019837 -0.14142573
##   bank_housing   bank_loan    bank_job bank_education  bank_month
## 1  -0.10792551 -0.04878717  0.07320974    0.003528136  0.26738813
## 2   0.89051694 -0.43679864  0.19137630   -0.020087393 -0.09697155
## 3   0.45966174 -0.10990489  0.14562605   -0.063929277 -0.50983956
## 4   0.04506119  2.28933378  0.15557965   -0.053291888  0.12667368
## 5  -0.25161852 -0.33781248 -0.27710759    0.132373273  0.48729757
## 6  -1.10768164 -0.43679864 -0.31025904    0.057696533  0.17595939
##   bank_contact bank_marital bank_default          day
## 1  -0.18423665  0.072315748   0.07513786  0.805075107
## 2  -0.32482319 -0.003034216  -0.01362707 -0.006344498
## 3   0.59371717 -0.055935749  -0.07985529 -0.284846774
## 4  -0.02384703  0.039959255   0.19509420  0.038076715
## 5   0.11084074  0.155767166  -0.12811250  0.034933748
## 6   0.15230242 -0.010660947  -0.03747494  0.018055664
```

```
##
## Clustering vector:
##     [1] 2 2 4 2 6 2 4 2 2 2 2 2 2 2 2 2 2 2 2 2 4 2 4 2 4 2 2 4 2 4 4 2 4
2
##    [35] 5 2 2 6 2 2 2 2 2 2 2 2 2 2 6 2 2 2 2 2 4 4 2 2 2 2 2 2 2 2 2 2 6
2
## [45119] 5 6 3 3 4 6 2 6 3 5 6 6 6 6 6 6 6 6 3 6 3 6 6 6 3 6 6 6 3 3 6 3 3
3
## [45153] 3 4 3 6 6 6 6 6 3 2 6 6 6 3 3 6 3 3 6 3 3 3 6 6 2 3 2 6 6 6 3 6 6
2
## [45187] 6 2 3 6 3 6 6 6 4 3 6 2 6 3 2 6 6 6 6 4 6 6 6 6 3
##
## Within cluster sum of squares by cluster:
## [1]  20189.48 140771.20  86071.80  73792.52  21111.20 157831.09
##  (between_SS / total_SS =  26.3 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
# Computing the percentage of variation accounted for. Six clusters
perc.var.6 <- round(100*(1 - kmeans6$betweenss/kmeans6$totss),1)
names(perc.var.6) <- "Perc. 6 clus"
perc.var.6
```

```
## Perc. 6 clus
##         73.7
```

```
(kmeans7 <- kmeans(bank_int_scale,7,nstart = 10))
```

```
## K-means clustering with 7 clusters of sizes 5131, 13328, 1265, 814, 10462,
6093, 8118
##
## Cluster means:
##           age     duration    campaign     balance      pdays    previous
## 1 -0.15530357 -0.016014953 -0.19402850 -0.04622250  2.4303498  1.34837297
## 2 -0.46712553  0.039154291 -0.15014301 -0.12456455 -0.3026953 -0.16845296
## 3 -0.04439200 -0.436759445  4.33165529 -0.08929343 -0.4019430 -0.23958281
## 4 -0.13050878 -0.072942418  0.11112496 -0.49311815 -0.2210303 -0.13513765
## 5 -0.11055222 -0.030593656 -0.11971990 -0.10101547 -0.3981537 -0.24114858
## 6 -0.02805758 -0.009392318 -0.10754689 -0.18959349 -0.3074597 -0.17805017
## 7  1.04861466  0.067688888 -0.08198331  0.56956582 -0.2104693 -0.08038068
##   bank_housing   bank_loan    bank_job bank_education  bank_month
## 1   0.44950118 -0.09198088  0.15915254   -0.060505888 -0.62088352
## 2   0.02261854 -0.43679864  0.08537735   -0.109561226  0.00225713
## 3  -0.14021993 -0.05320056  0.09302872   -0.001573986  0.27718687
## 4  -0.04562218  0.56791847  0.00116443   -0.050591261  0.11291829
## 5   0.42856428 -0.43679864  0.23428807   -0.068154441 -0.21541526
## 6   0.06615343  2.28933378  0.16677062   -0.060047754  0.12913878
## 7  -0.89677853 -0.42873912 -0.68248404    0.356339348  0.51489841
##   bank_contact bank_marital bank_default          day
## 1   0.58892633  -0.04938132   -0.1354884 -0.30964688
## 2   0.73712458  -0.18153253   -0.1354884  0.02422469
## 3  -0.15455455   0.07443245   -0.1295469  0.81211390
## 4  -0.07525334  -0.13513513    7.3805430  0.06739618
## 5  -1.35601488  -0.01744983   -0.1354884 -0.03510695
## 6  -0.02840866   0.04693329   -0.1354884  0.03893779
## 7   0.21807243   0.31846263   -0.1354884  0.03865335
##
## Clustering vector:
##     [1] 5 5 6 5 5 5 6 4 5 5 5 5 5 5 5 5 5 5 5 5 6 5 6 5 6 5 5 6 5 6 6 5 6
5
##    [35] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 6 5 5 5 5 5 5 5 5 5 5 7
5
##    [69] 5 6 6 6 6 5 5 5 6 5 6 4 5 5 5 5 6 5 5 5 6 5 5 6 5 5 5 6 5 5 5 5 5
5
##   [103] 5 5 5 5 5 6 5 5 5 5 6 5 5 5 5 5 5 7 5 5 5 6 6 5 6 6 5 5 5 5 5 6 5
5
## [45119] 7 7 1 1 6 7 2 2 1 7 2 7 2 7 7 2 7 7 1 2 1 7 7 7 1 2 7 2 1 2 2 1 1
1
## [45153] 7 7 1 7 7 7 7 7 2 2 2 7 7 1 1 7 7 1 7 1 1 1 7 7 2 1 2 2 7 7 1 7 7
7
## [45187] 7 2 1 2 2 7 2 2 6 7 2 2 7 1 2 7 2 2 7 6 7 7 7 7 1
##
## Within cluster sum of squares by cluster:
## [1]  80961.52 109220.25  17397.04  10207.64  77963.65  56392.48 105456.02
##  (between_SS / total_SS =  32.5 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
# Computing the percentage of variation accounted for. Six clusters
perc.var.7 <- round(100*(1 - kmeans7$betweenss/kmeans7$totss),1)
names(perc.var.7) <- "Perc. 7 clus"
perc.var.7
```

```
## Perc. 7 clus
##         67.5
```

```
perc.var.2
```

```
## Perc. 2 clus
##         92.1
```

```
perc.var.3
```

```
## Perc. 3 clus
##          85.4
```

```
perc.var.4
```

```
## Perc. 4 clus
##          80.3
```

```
perc.var.5
```

```
## Perc. 5 clus
##          75.2
```

```
perc.var.6
```

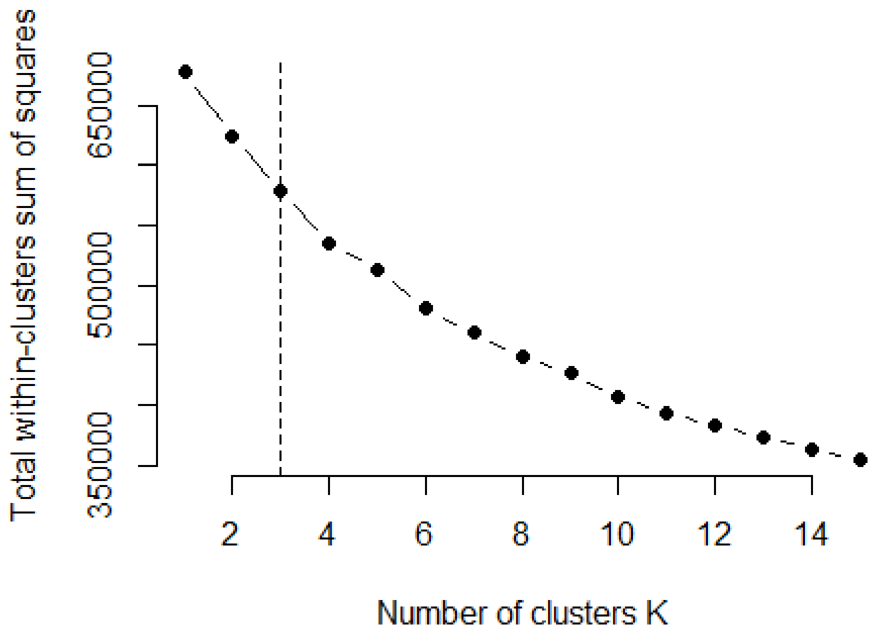
```
## Perc. 6 clus
##          73.7
```

```
perc.var.7
```

```
## Perc. 7 clus
##          67.5
```

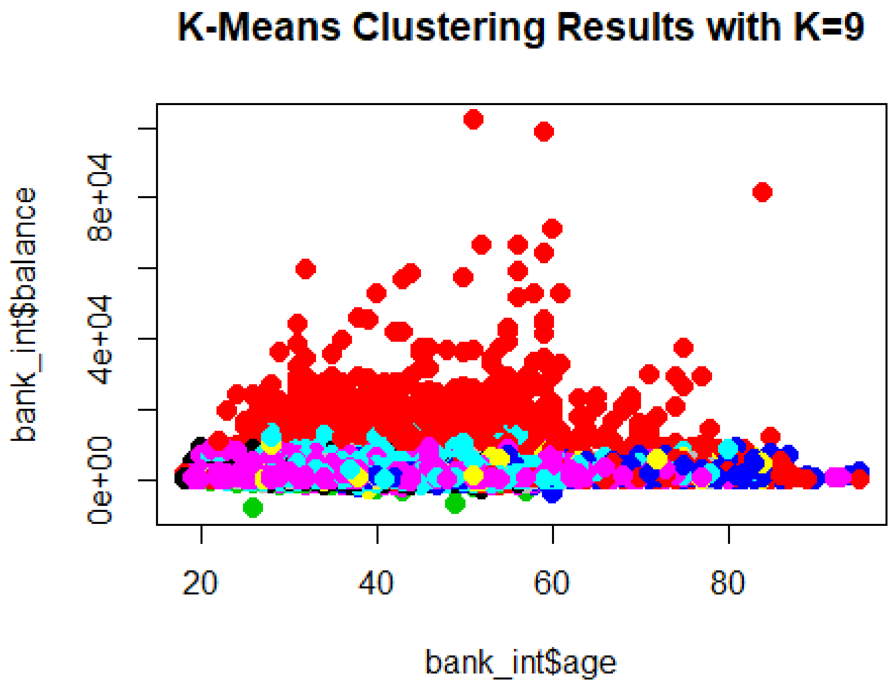
```
k.max <- 15 # Maximal number of clusters
wss <- sapply(1:k.max,
              function(k){kmeans(bank_int_scale, k, nstart=50 )$tot.withinss}
)
```

```
plot(1:k.max, wss,
     type="b", pch = 19, frame = FALSE,
     xlab="Number of clusters K",
     ylab="Total within-clusters sum of squares")
abline(v = 3, lty =2)
```

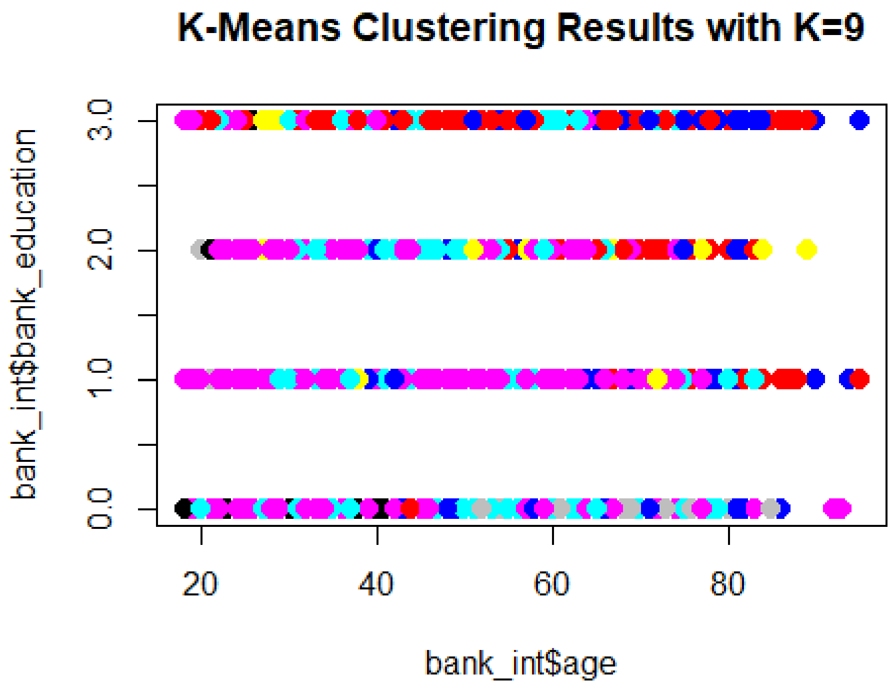


```
(kmeans9 <- kmeans(bank_int_scale,9,nstart = 10))

## K-means clustering with 9 clusters of sizes 5424, 815, 4294, 4845, 10270,
2222, 8804, 7682, 855
##
## Cluster means:
##           age    duration    campaign      balance       pdays    previous
## 1  0.72365922 -0.19654029  0.01177887 -0.050322628 -0.34130104 -0.19727075
## 2 -0.13195897 -0.07396383  0.12415204 -0.492614763 -0.22126396 -0.13528096
## 3  0.52286186 -0.14083272 -0.02976281 -0.122809060 -0.28689536 -0.16316980
## 4 -0.13088850 -0.08273165 -0.19152732 -0.095759057  2.47759559  1.37939930
## 5  0.04783949 -0.22293234  0.25179742 -0.034571341 -0.20697665 -0.09603857
## 6 -0.02989033  3.14825457 -0.04666809  0.005833961 -0.21171021 -0.12064261
## 7 -0.19444716 -0.18940260  0.03487251 -0.132762870 -0.39548546 -0.23817985
## 8 -0.57215964 -0.11309182 -0.24185154 -0.153301695 -0.33795817 -0.18723016
## 9  0.29678686 -0.04415095 -0.04760019  5.092769873 -0.07760935 -0.01481459
##   bank_housing    bank_loan     bank_job bank_education  bank_month
## 1  -0.09419672 -0.095026875 -0.087561354     1.54286669 -0.06792657
## 2  -0.04446938  0.570030636  0.002665732    -0.05138757  0.11323447
## 3  -0.05847841  0.047607285 -0.023259025    -0.01590247  0.11561053
## 4   0.44902224 -0.027175439  0.169106376    -0.06899174 -0.64922759
## 5  -0.57074175  0.016848908 -0.190231535    -0.11795441  0.94136520
## 6   0.01442632 -0.034381252 -0.052151664     0.03178003  0.08042462
## 7   0.37407614  0.019311431  0.168804780    -0.56300413 -0.19377374
## 8   0.18523664 -0.001724654  0.076359110    -0.24964187 -0.72582124
## 9  -0.30892706 -0.322014116 -0.292251502     0.10659172  0.42156994
##   bank_contact bank_marital bank_default          day
## 1  -0.61805705  0.612683366   -0.1354884 -0.111229331
## 2  -0.07425617 -0.135831222    7.3805430  0.069553492
## 3  -0.10647562 -2.124386657   -0.1354884  0.059721154
## 4   0.58188799 -0.017383499   -0.1354884 -0.315683094
## 5   0.69300125  0.343557346   -0.1354884  0.398857483
## 6   0.07228200  0.009361813   -0.1354884 -0.009942525
## 7  -1.35601488  0.243175753   -0.1354884  0.007783275
## 8   0.73226095  0.022109465   -0.1354884 -0.304957210
## 9   0.10083295  0.156663265   -0.1354884  0.022978982
##
## Clustering vector:
##     [1] 1 7 7 7 7 7 7 2 1 7 3 7 7 7 7 1 7 1 1 7 7 1 1 7 1 7 7 7 7 7 7 1 7
7
##    [35] 9 3 7 6 3 7 3 7 7 6 7 1 1 7 3 7 7 3 7 6 7 7 7 7 7 6 7 3 1 7 7 1 1
3
##    [69] 7 9 7 7 7 1 7 7 7 3 3 2 7 3 7 6 1 1 6 6 3 7 7 7 7 7 3 7 7 1 7 1 7
4
## [45119] 9 5 4 4 7 5 6 3 4 9 5 5 5 5 5 5 5 1 4 5 4 5 5 5 4 5 3 5 4 3 5 5 4
4
## [45153] 4 5 4 5 5 5 5 5 5 5 5 5 5 4 4 5 5 4 5 4 5 5 5 5 5 4 5 5 5 5 5 1 5
5
## [45187] 5 5 4 5 5 3 5 5 4 5 5 5 5 4 6 5 5 5 5 5 6 3 6 5 4
##
## Within cluster sum of squares by cluster:
## [1] 47712.98 10332.42 41440.79 75424.75 93113.27 27810.43 66780.70 58825.9
1
## [9] 18386.10
##  (between_SS / total_SS =  35.1 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

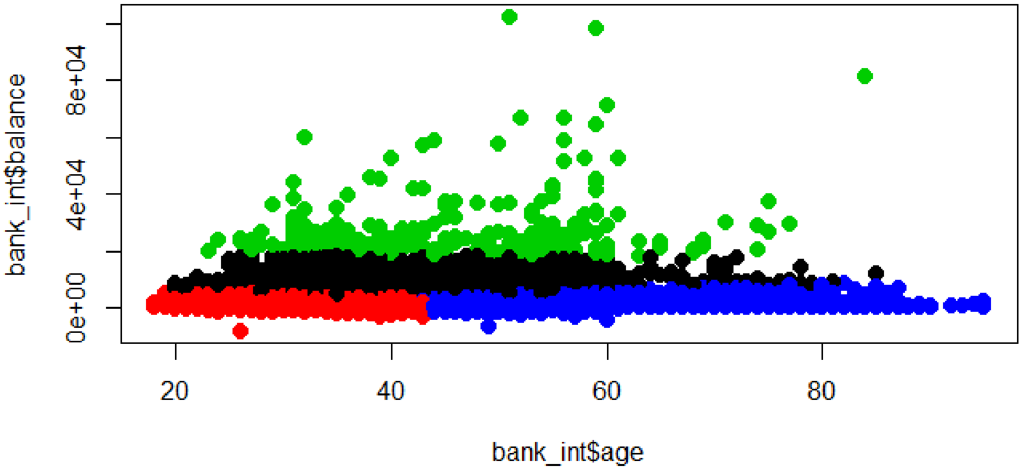
```
plot(bank_int$age, bank_int$balance,col=(kmeans9$cluster+1), main="K-Means Cl
ustering Results with K=9",  pch=20,cex=2)
```



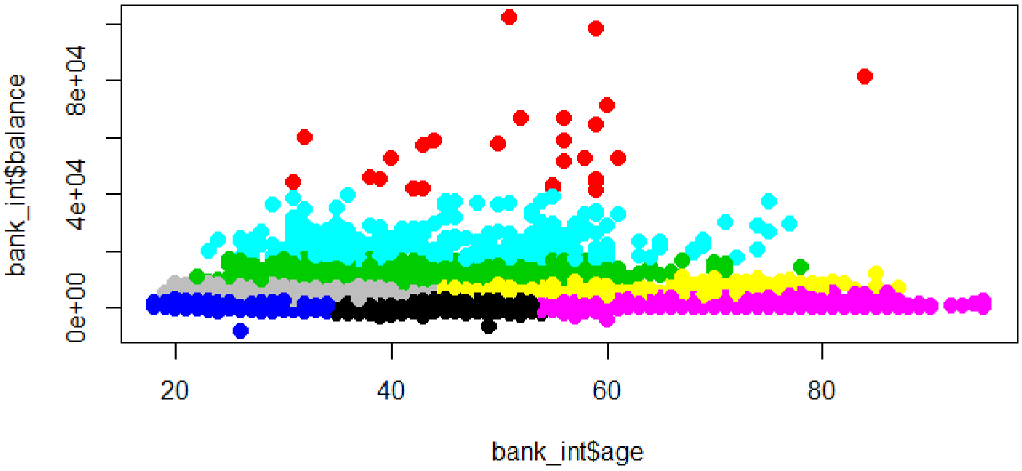
```
plot(bank_int$age, bank_int$bank_education,col=(kmeans9$cluster+1), main="K-M
eans Clustering Results with K=9",  pch=20,cex=2)
```



```
feat.scaled <- scale(bank_int[,c("age","balance")])
set.seed(15555)
pclusters <- kmeans(feat.scaled, 4, nstart=20, iter.max=100)

groups <- pclusters$cluster
#clusterDF <- cbind(as.data.frame(feat.scaled), Cluster=as.factor(groups)

plot(bank_int$age, bank_int$balance, col=groups)
```



```
pclusters <- kmeans(feat.scaled, 9, nstart=20, iter.max=100)

groups <- pclusters$cluster
#clusterDF <- cbind(as.data.frame(feat.scaled), Cluster=as.factor(groups))

plot(bank_int$age, bank_int$balance, col=groups)
```


bank_int$balance
0e+00
4e+04
8e+04
bank_int$age