Final Project

Brandi Rodriguez

May 9, 2021

LOAD LIBRARIES

```
library(RSADBE) #dataset source
library(dplyr)
library(tidyr) # data wrangling
library(ggplot2) # plotting
library(survival) # survival
library(rpart) # DT
library(randomForest) #RF
library(randomForestSRC) # RF
```

LOAD DATA

https://cran.r-project.org/web/packages/RSADBE/RSADBE.pdf

```
library(RSADBE)
data(GC)
str(GC)
```

```
## 'data.frame':    1000 obs. of  21 variables:
##  $ checking: int  1 2 4 1 1 4 4 2 4 2 ...
##  $ duration: int  6 48 12 42 24 36 24 36 12 30 ...
##  $ history : int  4 2 4 2 3 2 2 2 2 4 ...
##  $ purpose : Factor w/ 10 levels "0","1","2","3",..: 4 4 7 3 1 7 3 2 4 1 ...
##  $ amount  : num  1169 5951 2096 7882 4870 ...
##  $ savings : int  5 1 1 1 1 5 3 1 4 1 ...
##  $ employed: int  5 3 4 4 3 3 5 3 4 1 ...
##  $ installp: int  4 2 2 2 3 2 3 2 2 4 ...
##  $ marital : int  3 2 3 3 3 3 3 3 1 4 ...
##  $ coapp   : int  1 1 1 3 1 1 1 1 1 1 ...
##  $ resident: int  4 2 3 4 4 4 4 2 4 2 ...
##  $ property: Factor w/ 4 levels "1","2","3","4": 1 1 1 2 4 4 2 3 1 3 ...
##  $ age     : num  67 22 49 45 53 35 53 35 61 28 ...
##  $ other   : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ housing : int  2 2 2 3 3 3 2 1 2 2 ...
##  $ existcr : int  2 1 1 1 2 1 1 1 1 2 ...
##  $ job     : int  3 3 2 3 3 2 3 4 2 4 ...
```

```
##  $ depends : int  1 1 2 2 2 2 1 1 1 1 ...
##  $ telephon: int  2 1 1 1 1 2 1 2 1 1 ...
##  $ foreign : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ good_bad: Factor w/ 2 levels "bad","good": 2 1 2 2 1 2 2 2 2 1 ...
```

```
summary(GC)
```

```
##     checking        duration        history         purpose        amount     
##  Min.   :1.000   Min.   : 4.0   Min.   :0.000   3      :280   Min.   :  250  
##  1st Qu.:1.000   1st Qu.:12.0   1st Qu.:2.000   0      :234   1st Qu.: 1366  
##  Median :2.000   Median :18.0   Median :2.000   2      :181   Median : 2320  
##  Mean   :2.577   Mean   :20.9   Mean   :2.545   1      :103   Mean   : 3271  
##  3rd Qu.:4.000   3rd Qu.:24.0   3rd Qu.:4.000   9      : 97   3rd Qu.: 3972  
##  Max.   :4.000   Max.   :72.0   Max.   :4.000   6      : 50   Max.   :18424  
##                                                 (Other): 55                  
##     savings         employed        installp        marital     
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000  
##  1st Qu.:1.000   1st Qu.:3.000   1st Qu.:2.000   1st Qu.:2.000  
##  Median :1.000   Median :3.000   Median :3.000   Median :3.000  
##  Mean   :2.105   Mean   :3.384   Mean   :2.973   Mean   :2.682  
##  3rd Qu.:3.000   3rd Qu.:5.000   3rd Qu.:4.000   3rd Qu.:3.000  
##  Max.   :5.000   Max.   :5.000   Max.   :4.000   Max.   :4.000  
##                                                                 
##      coapp          resident     property      age            other      
##  Min.   :1.000   Min.   :1.000   1:282    Min.   :19.00   Min.   :1.000  
##  1st Qu.:1.000   1st Qu.:2.000   2:232    1st Qu.:27.00   1st Qu.:3.000  
##  Median :1.000   Median :3.000   3:332    Median :33.00   Median :3.000  
##  Mean   :1.145   Mean   :2.845   4:154    Mean   :35.55   Mean   :2.675  
##  3rd Qu.:1.000   3rd Qu.:4.000            3rd Qu.:42.00   3rd Qu.:3.000  
##  Max.   :3.000   Max.   :4.000            Max.   :75.00   Max.   :3.000  
##                                                                          
##     housing         existcr           job           depends     
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000  
##  1st Qu.:2.000   1st Qu.:1.000   1st Qu.:3.000   1st Qu.:1.000  
##  Median :2.000   Median :1.000   Median :3.000   Median :1.000  
##  Mean   :1.929   Mean   :1.407   Mean   :2.904   Mean   :1.155  
##  3rd Qu.:2.000   3rd Qu.:2.000   3rd Qu.:3.000   3rd Qu.:1.000  
##  Max.   :3.000   Max.   :4.000   Max.   :4.000   Max.   :2.000  
##                                                                 
##     telephon        foreign      good_bad  
##  Min.   :1.000   Min.   :1.000   bad :300  
##  1st Qu.:1.000   1st Qu.:1.000   good:700  
##  Median :1.000   Median :1.000             
##  Mean   :1.404   Mean   :1.037             
##  3rd Qu.:2.000   3rd Qu.:1.000             
##  Max.   :2.000   Max.   :2.000             
## 
```

DATA PREPROCESSING

```r
#create copy of dataset
df = GC

#rename variables and recode response variable
df = df %>%
  rename(response = good_bad,
         dependents = depends,
         telephone = telephon) %>%
  mutate(response = recode(response, "bad" = 1, "good" = 0))

#convert factor variables
factors = c("checking", "history", "savings", "employed", "marital", "coapp", "other", "housing", "job"
df[factors] = lapply(df[factors], factor)

#view data
str(df)
```

```
## 'data.frame':    1000 obs. of  21 variables:
##  $ checking  : Factor w/ 4 levels "1","2","3","4": 1 2 4 1 1 4 4 2 4 2 ...
##  $ duration  : int  6 48 12 42 24 36 24 36 12 30 ...
##  $ history   : Factor w/ 5 levels "0","1","2","3",..: 5 3 5 3 4 3 3 3 3 5 ...
##  $ purpose   : Factor w/ 10 levels "0","1","2","3",..: 4 4 7 3 1 7 3 2 4 1 ...
##  $ amount    : num  1169 5951 2096 7882 4870 ...
##  $ savings   : Factor w/ 5 levels "1","2","3","4",..: 5 1 1 1 1 5 3 1 4 1 ...
##  $ employed  : Factor w/ 5 levels "1","2","3","4",..: 5 3 4 4 3 3 5 3 4 1 ...
##  $ installp  : int  4 2 2 2 3 2 3 2 2 4 ...
##  $ marital   : Factor w/ 4 levels "1","2","3","4": 3 2 3 3 3 3 3 3 1 4 ...
##  $ coapp     : Factor w/ 3 levels "1","2","3": 1 1 1 3 1 1 1 1 1 1 ...
##  $ resident  : int  4 2 3 4 4 4 4 2 4 2 ...
##  $ property  : Factor w/ 4 levels "1","2","3","4": 1 1 1 2 4 4 2 3 1 3 ...
##  $ age       : num  67 22 49 45 53 35 53 35 61 28 ...
##  $ other     : Factor w/ 3 levels "1","2","3": 3 3 3 3 3 3 3 3 3 3 ...
##  $ housing   : Factor w/ 3 levels "1","2","3": 2 2 2 3 3 3 2 1 2 2 ...
##  $ existcr   : int  2 1 1 1 2 1 1 1 1 2 ...
##  $ job       : Factor w/ 4 levels "1","2","3","4": 3 3 2 3 3 2 3 4 2 4 ...
##  $ dependents: int  1 1 2 2 2 2 1 1 1 1 ...
##  $ telephone : Factor w/ 2 levels "1","2": 2 1 1 1 1 2 1 2 1 1 ...
##  $ foreign   : Factor w/ 2 levels "1","2": 1 1 1 1 1 1 1 1 1 1 ...
##  $ response  : Factor w/ 2 levels "0","1": 1 2 1 1 2 1 1 1 1 2 ...
```

```r
summary(GC)
```

```
##     checking        duration       history          purpose        amount
##  Min.   :1.000   Min.   : 4.0   Min.   :0.000   3      :280   Min.   :  250
##  1st Qu.:1.000   1st Qu.:12.0   1st Qu.:2.000   0      :234   1st Qu.: 1366
##  Median :2.000   Median :18.0   Median :2.000   2      :181   Median : 2320
##  Mean   :2.577   Mean   :20.9   Mean   :2.545   1      :103   Mean   : 3271
##  3rd Qu.:4.000   3rd Qu.:24.0   3rd Qu.:4.000   9      : 97   3rd Qu.: 3972
##  Max.   :4.000   Max.   :72.0   Max.   :4.000   6      : 50   Max.   :18424
##                                                 (Other): 55
##     savings         employed        installp        marital
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
```

```
##  1st Qu.:1.000   1st Qu.:3.000   1st Qu.:2.000   1st Qu.:2.000
##  Median :1.000   Median :3.000   Median :3.000   Median :3.000
##  Mean   :2.105   Mean   :3.384   Mean   :2.973   Mean   :2.682
##  3rd Qu.:3.000   3rd Qu.:5.000   3rd Qu.:4.000   3rd Qu.:3.000
##  Max.   :5.000   Max.   :5.000   Max.   :4.000   Max.   :4.000
##
##      coapp          resident      property      age            other
##  Min.   :1.000   Min.   :1.000   1:282    Min.   :19.00   Min.   :1.000
##  1st Qu.:1.000   1st Qu.:2.000   2:232    1st Qu.:27.00   1st Qu.:3.000
##  Median :1.000   Median :3.000   3:332    Median :33.00   Median :3.000
##  Mean   :1.145   Mean   :2.845   4:154    Mean   :35.55   Mean   :2.675
##  3rd Qu.:1.000   3rd Qu.:4.000            3rd Qu.:42.00   3rd Qu.:3.000
##  Max.   :3.000   Max.   :4.000            Max.   :75.00   Max.   :3.000
##
##     housing         existcr           job           depends
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:1.000   1st Qu.:3.000   1st Qu.:1.000
##  Median :2.000   Median :1.000   Median :3.000   Median :1.000
##  Mean   :1.929   Mean   :1.407   Mean   :2.904   Mean   :1.155
##  3rd Qu.:2.000   3rd Qu.:2.000   3rd Qu.:3.000   3rd Qu.:1.000
##  Max.   :3.000   Max.   :4.000   Max.   :4.000   Max.   :2.000
##
##     telephon        foreign       good_bad
##  Min.   :1.000   Min.   :1.000   bad :300
##  1st Qu.:1.000   1st Qu.:1.000   good:700
##  Median :1.000   Median :1.000
##  Mean   :1.404   Mean   :1.037
##  3rd Qu.:2.000   3rd Qu.:1.000
##  Max.   :2.000   Max.   :2.000
##
```

MISSING VALUES

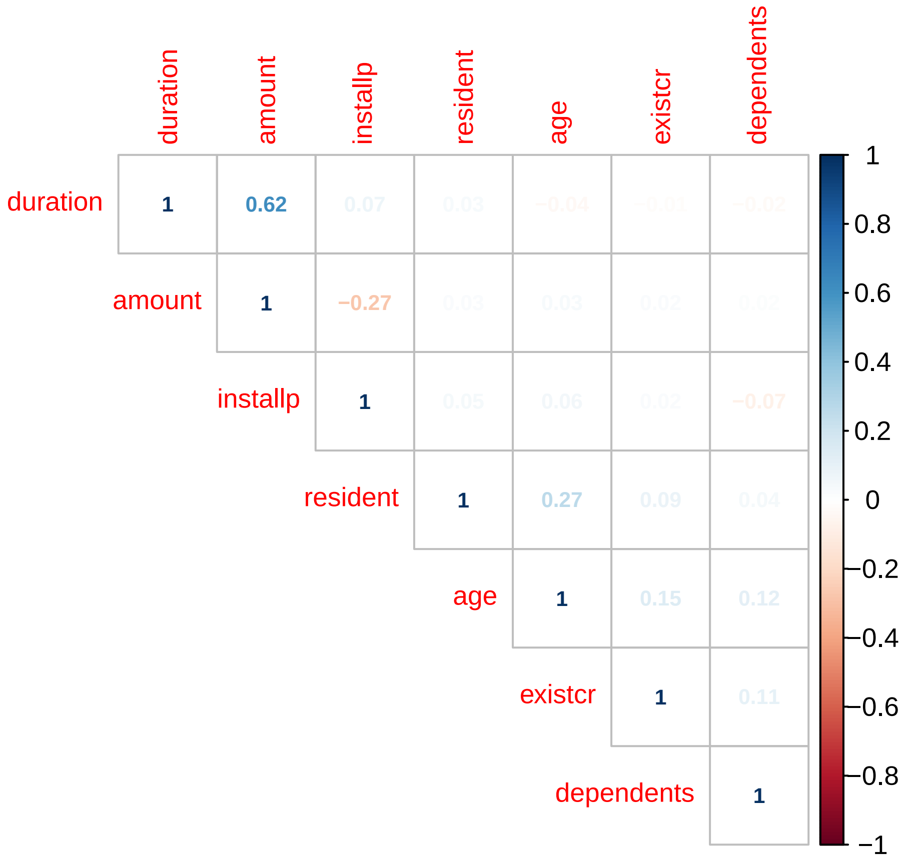
This was a complete dataset with no missing values

```r
colSums(is.na(df))
```

```
##   checking   duration    history    purpose     amount    savings   employed
##          0          0          0          0          0          0          0
##   installp    marital      coapp   resident   property        age      other
##          0          0          0          0          0          0          0
##    housing    existcr        job dependents  telephone    foreign   response
##          0          0          0          0          0          0          0
```

CORRELATIONS

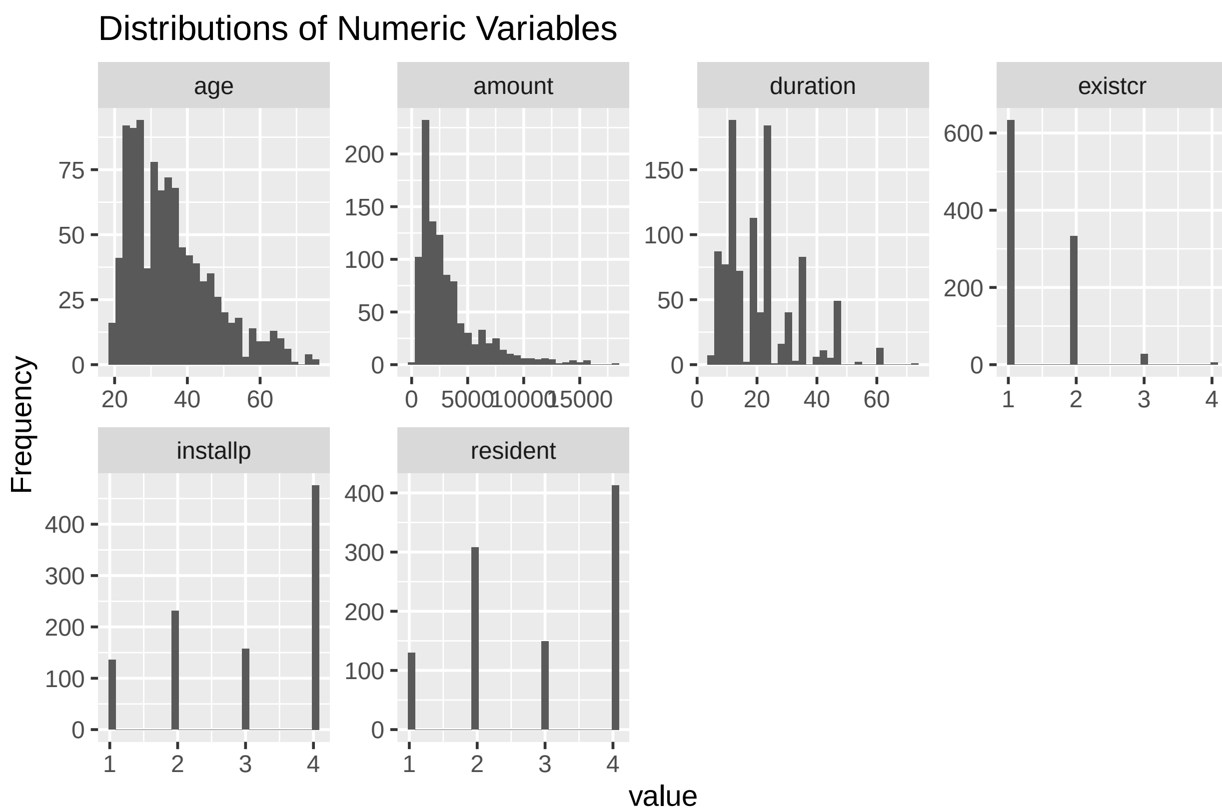
```r
library(corrplot)
corrplot(cor(df[sapply(df, is.numeric)]), method = "number", type = "upper", tl.cex = .80, number.cex =
```



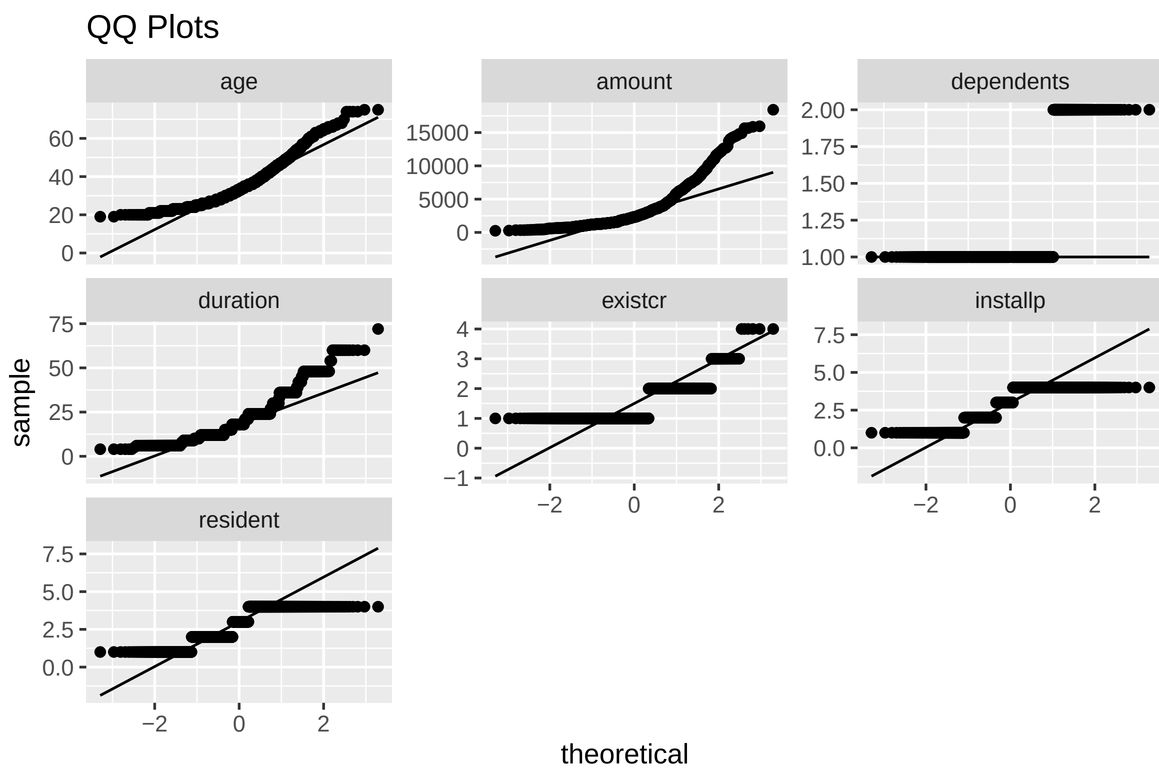
EDA

```r
library(DataExplorer)
library(ggplot2)
```

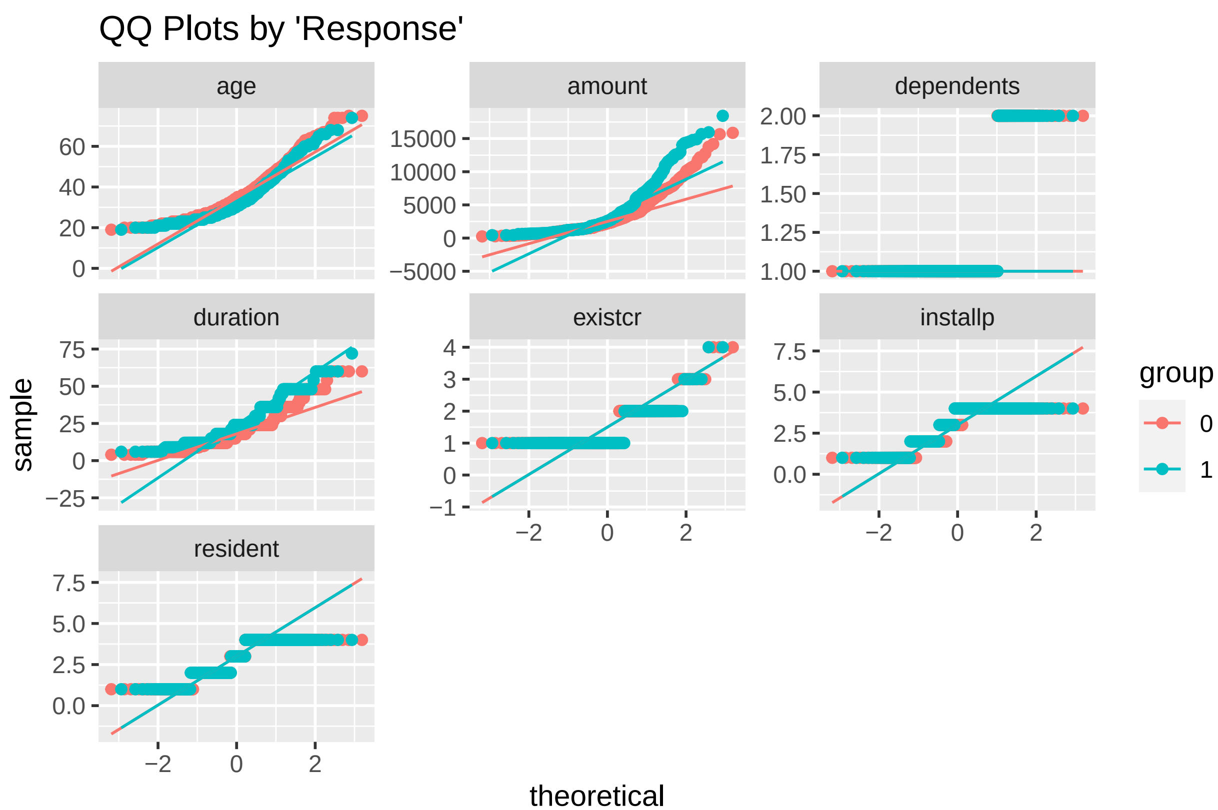
```r
plot_histogram(df, title = "Distributions of Numeric Variables")
```



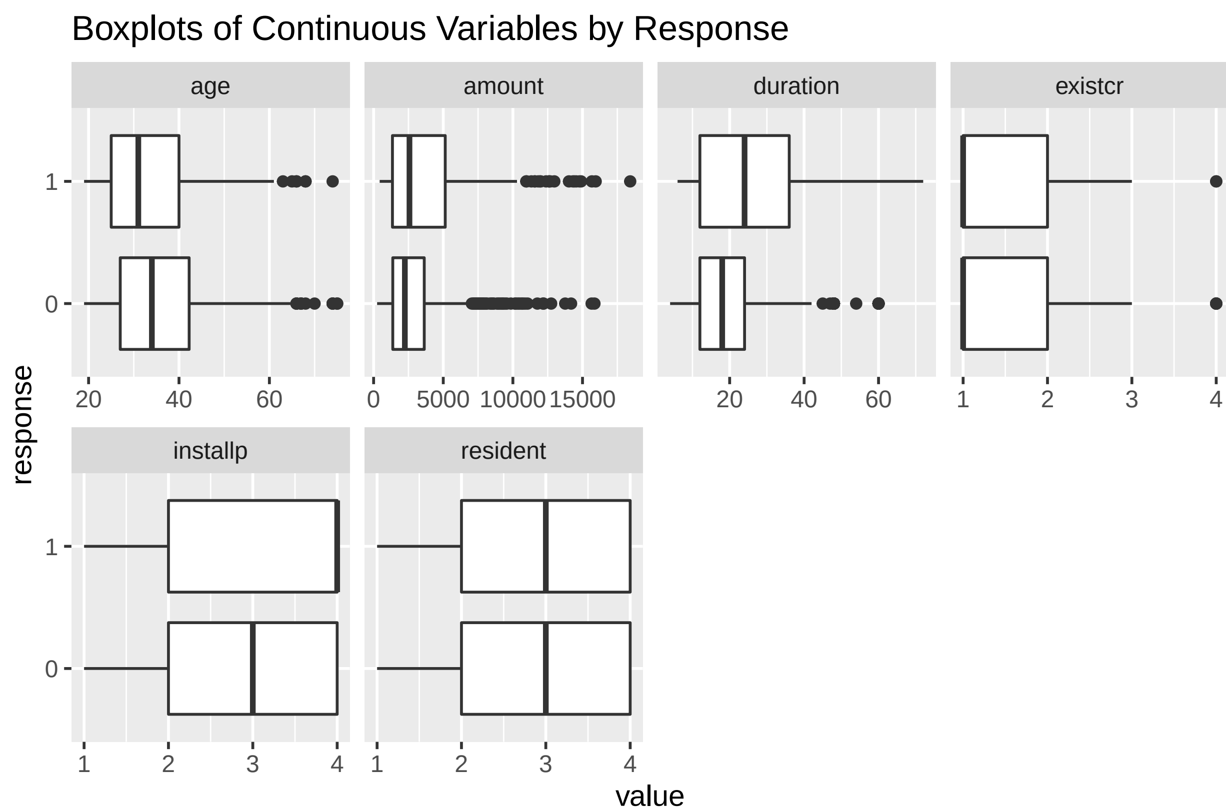

```r
plot_qq(df, title="QQ Plots")
```



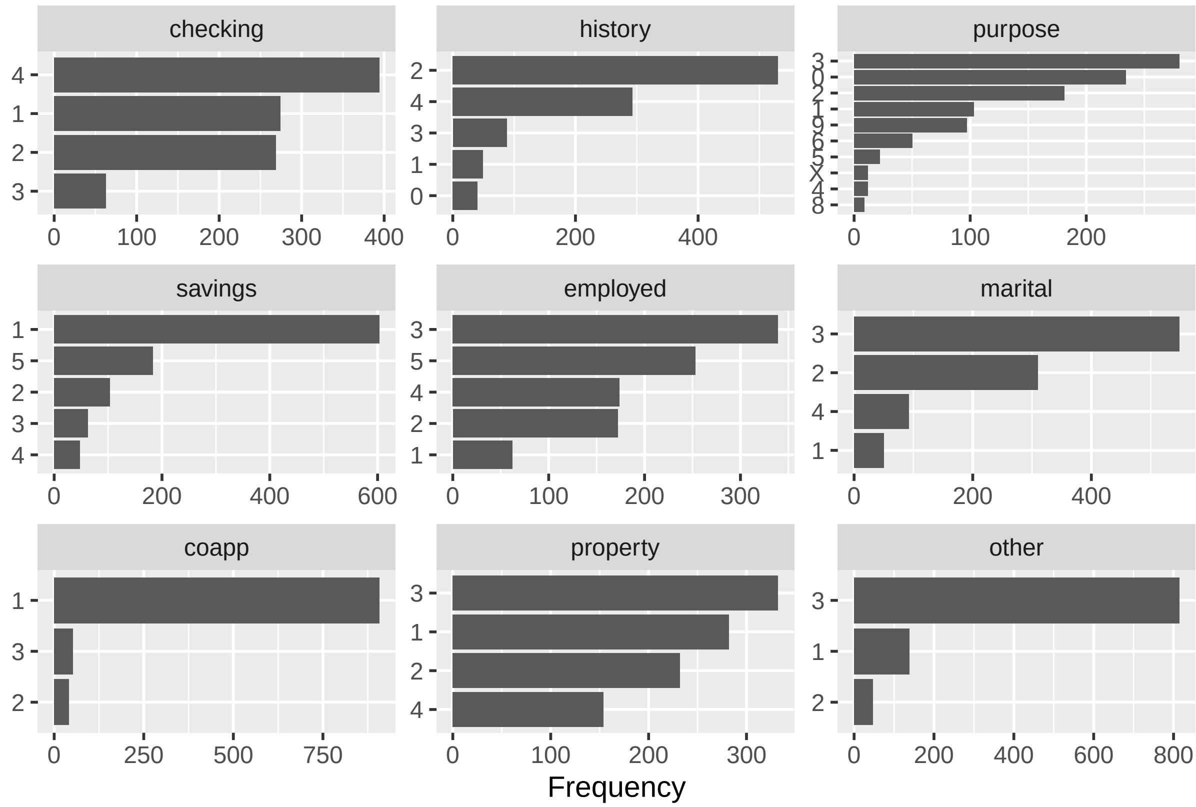

```r
plot_qq(df, by = "response",
        title = "QQ Plots by 'Response'")
```

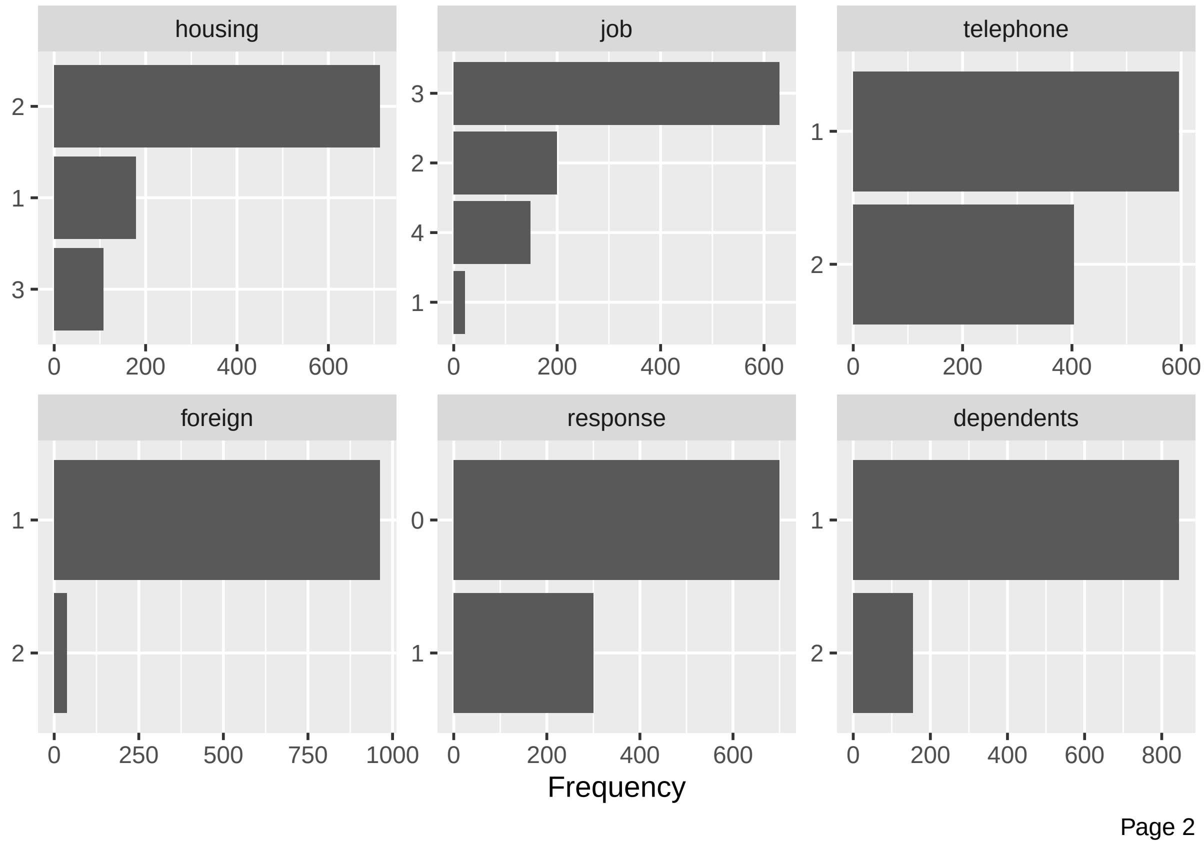



```r
plot_boxplot(df, by = "response", title = "Boxplots of Continuous Variables by Response")
```

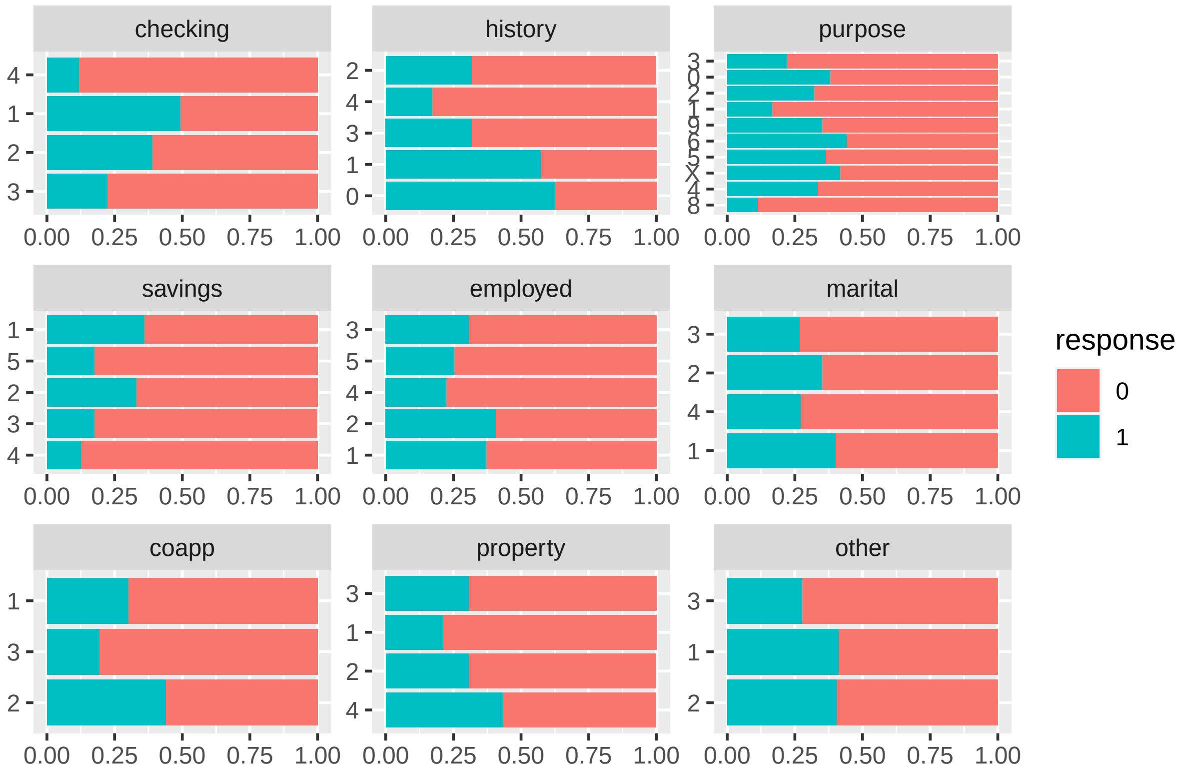

Boxplots of Continuous Variables by Response
age
amount
duration
existcr
installp
resident
response
value
1
0
20
40
60
0
5000
10000
15000
1
2
3
4

```
plot_bar(df)
```

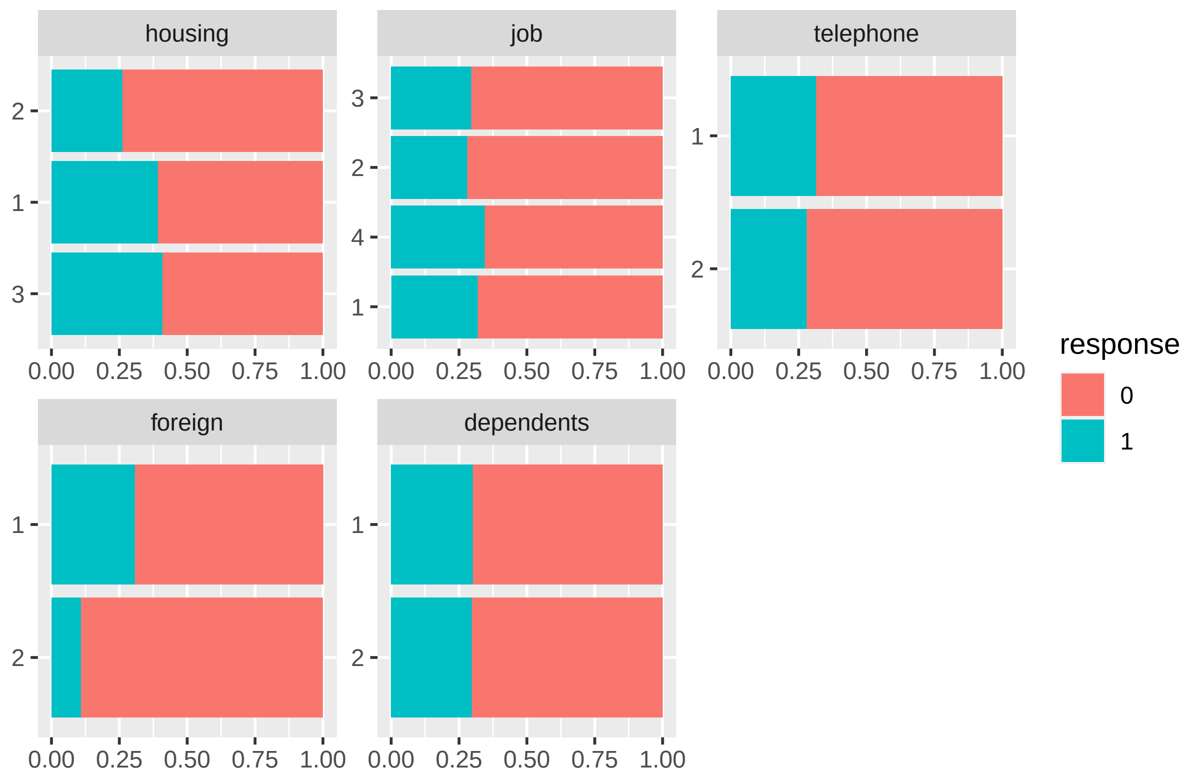

checking
history
purpose
savings
employed
marital
coapp
property
other
Frequency




```r
plot_bar(df, by = "response")
```


checking
history
purpose
savings
employed
marital
coapp
property
other
response
0
1
0.00
0.25
0.50
0.75
1.00

Page 1



Page 2

RENAME FACTOR LEVELS

http://www1.beuth-hochschule.de/FB_II/reports/Report-2019-004.pdf

```r
prop.table(table(df$dependents, df$response), margin=2)*100
```

```
##
##            0        1
##   1 84.42857 84.66667
##   2 15.57143 15.33333
```

```r
library(tidyverse)
levels(df$housing) = c("free", "rent", "own")
levels(df$checking) = c("no checking account", "<0", "<200","200+/salary for atleast 1 year")
levels(df$history) = c("delayed previously", "critical/other existing credit", "no credits taken/all pa
levels(df$purpose) = c("others", "car (new)", "car (used)", "furniture/equipment", "radio/tv", "applian
levels(df$savings) = c("unknown/none", "<100", "<500", "<1000", "1000+")
levels(df$employed) = c("unemployed", "<1", "<4", "<7", "7+")
levels(df$marital) = c("male: divorced/separated", "female: non-single or male: single", "male: married
levels(df$coapp) = c("none", "co-applicant", "guarantor")
levels(df$property) = c("unknown/no property", "car or other", "building soc. savings agr./life ins.",
levels(df$other) = c("bank", "stores", "none")
levels(df$job) = c("unemployed/unskilled - non-resident", "unskilled - resident", "skilled employee/off
```

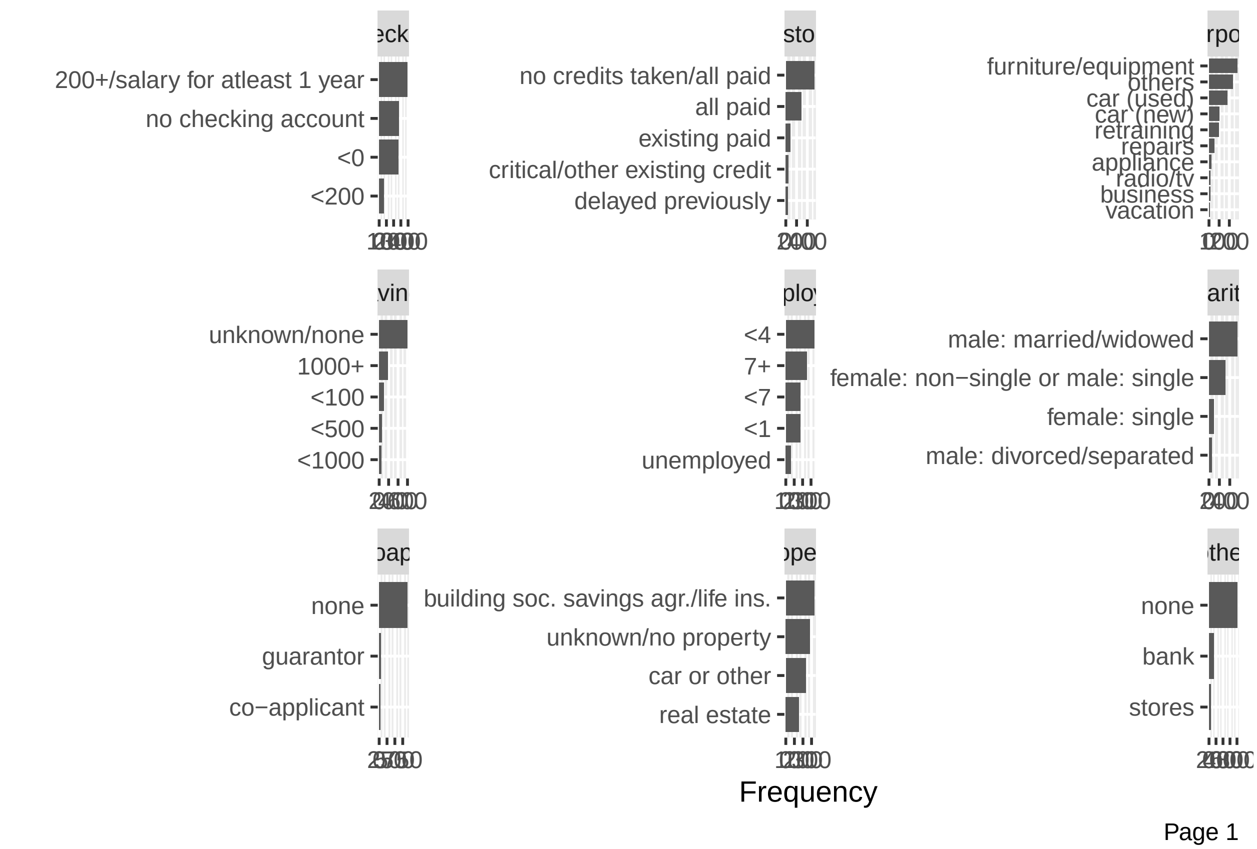
```r
levels(df$telephone) = c("no", "yes")
levels(df$foreign) = c("no", "yes")
levels(df$dependents) = c("0 to 2", "3+")
```

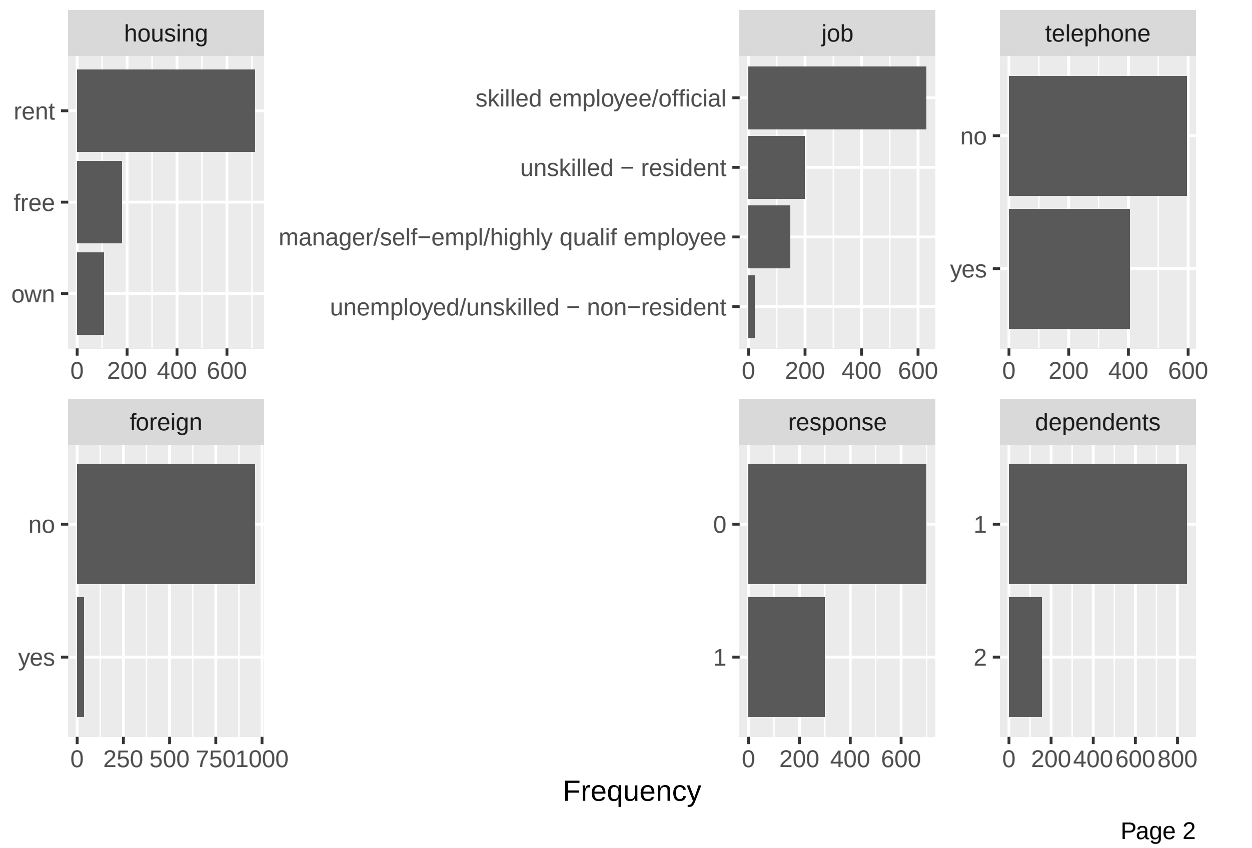
```r
#validate distribution tables match as defined in http://www1.beuth-hochschule.de/FB_II/reports/Report-

prop.table(table(df$housing, df$response), margin=2)*100
```

```
##
##                0         1
##   free 15.571429 23.333333
##   rent 75.285714 62.000000
##   own   9.142857 14.666667
```

```r
plot_bar(df)
```

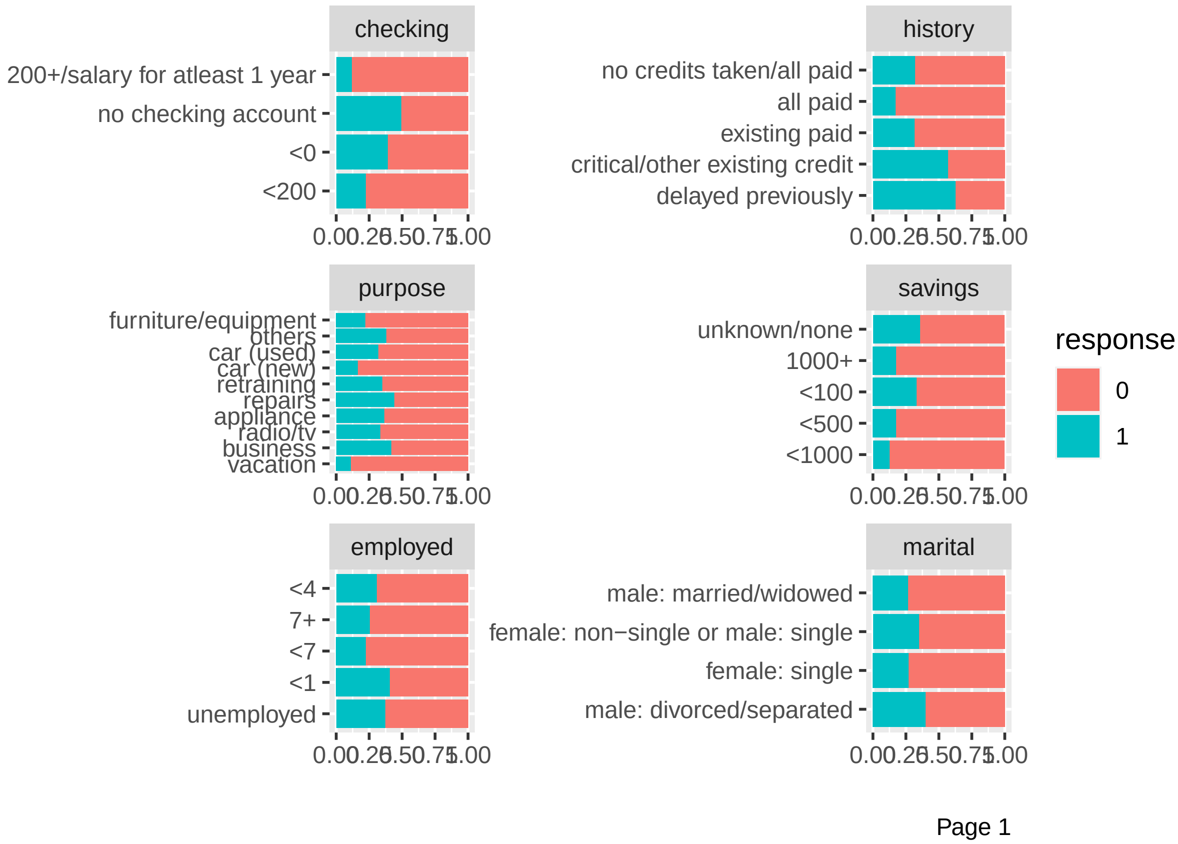


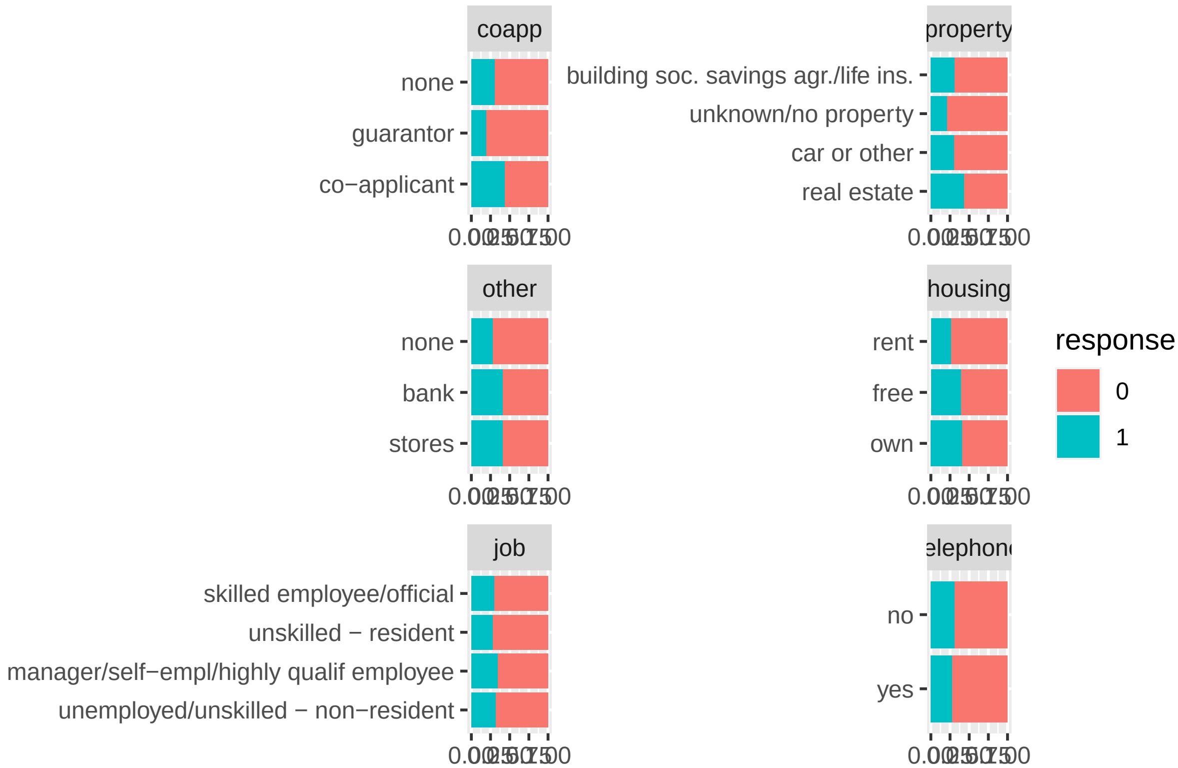



```r
str(df)
```

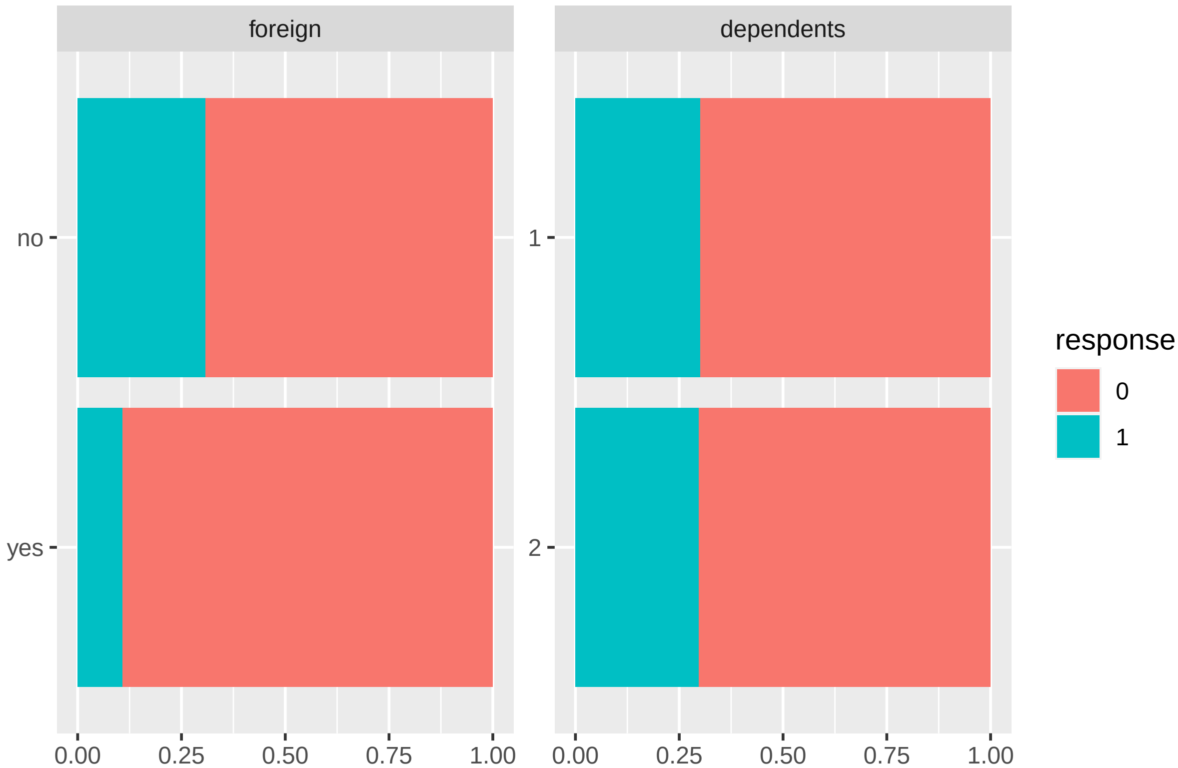
```
## 'data.frame':    1000 obs. of  21 variables:
##  $ checking  : Factor w/ 4 levels "no checking account",..: 1 2 4 1 1 4 4 2 4 2 ...
##  $ duration  : int  6 48 12 42 24 36 24 36 12 30 ...
##  $ history   : Factor w/ 5 levels "delayed previously",..: 5 3 5 3 4 3 3 3 3 5 ...
##  $ purpose   : Factor w/ 10 levels "others","car (new)",..: 4 4 7 3 1 7 3 2 4 1 ...
##  $ amount    : num  1169 5951 2096 7882 4870 ...
##  $ savings   : Factor w/ 5 levels "unknown/none",..: 5 1 1 1 1 5 3 1 4 1 ...
##  $ employed  : Factor w/ 5 levels "unemployed","<1",..: 5 3 4 4 3 3 5 3 4 1 ...
##  $ installp  : int  4 2 2 2 3 2 3 2 2 4 ...
##  $ marital   : Factor w/ 4 levels "male: divorced/separated",..: 3 2 3 3 3 3 3 3 1 4 ...
##  $ coapp     : Factor w/ 3 levels "none","co-applicant",..: 1 1 1 3 1 1 1 1 1 1 ...
##  $ resident  : int  4 2 3 4 4 4 4 2 4 2 ...
##  $ property  : Factor w/ 4 levels "unknown/no property",..: 1 1 1 2 4 4 2 3 1 3 ...
##  $ age       : num  67 22 49 45 53 35 53 35 61 28 ...
##  $ other     : Factor w/ 3 levels "bank","stores",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ housing   : Factor w/ 3 levels "free","rent",..: 2 2 2 3 3 3 2 1 2 2 ...
##  $ existcr   : int  2 1 1 1 2 1 1 1 1 2 ...
##  $ job       : Factor w/ 4 levels "unemployed/unskilled - non-resident",..: 3 3 2 3 3 2 3 4 2 4 ...
##  $ dependents: int  1 1 2 2 2 2 1 1 1 1 ...
##   ..- attr(*, "levels")= chr  "0 to 2" "3+"
##  $ telephone : Factor w/ 2 levels "no","yes": 2 1 1 1 1 2 1 2 1 1 ...
##  $ foreign   : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ response  : Factor w/ 2 levels "0","1": 1 2 1 1 2 1 1 1 1 2 ...
```

```r
plot_bar(df, by = "response", ncol = 2)
```



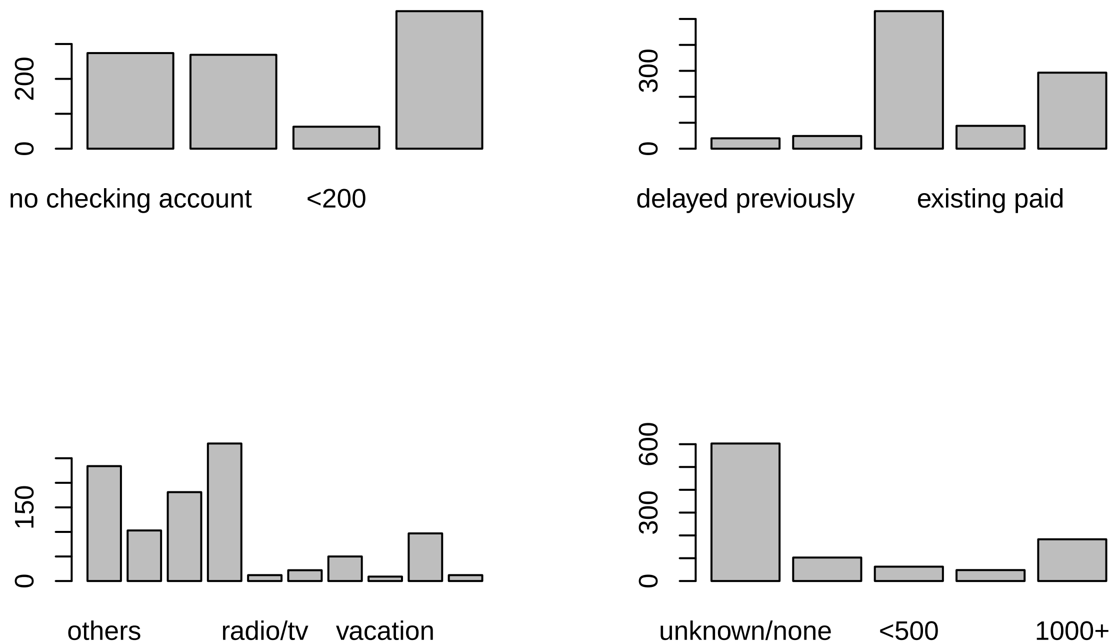

coapp
none
guarantor
co−applicant
property
building soc. savings agr./life ins.
unknown/no property
car or other
real estate
other
none
bank
stores
housing
rent
free
own
response
0
1
job
skilled employee/official
unskilled − resident
manager/self−empl/highly qualif employee
unemployed/unskilled − non−resident
elephon
no
yes

Page 2

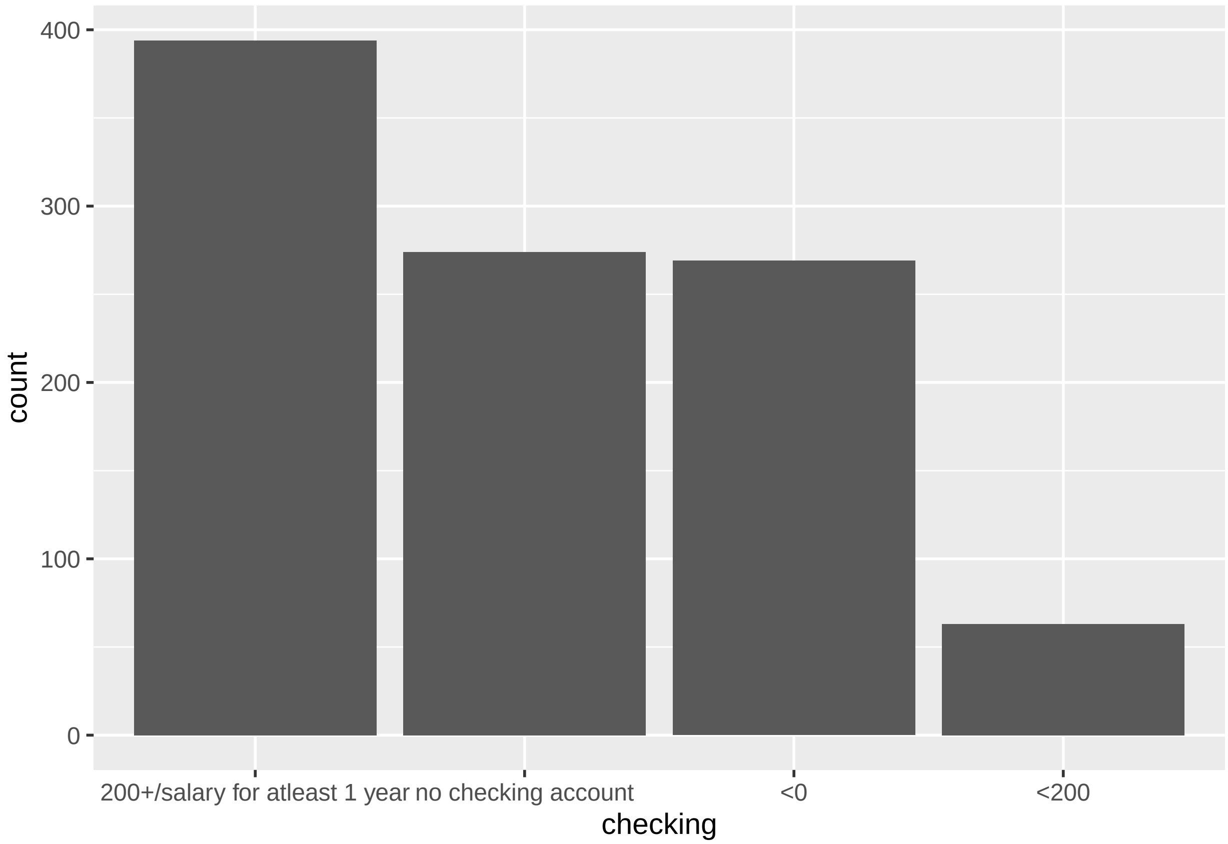


Page 3

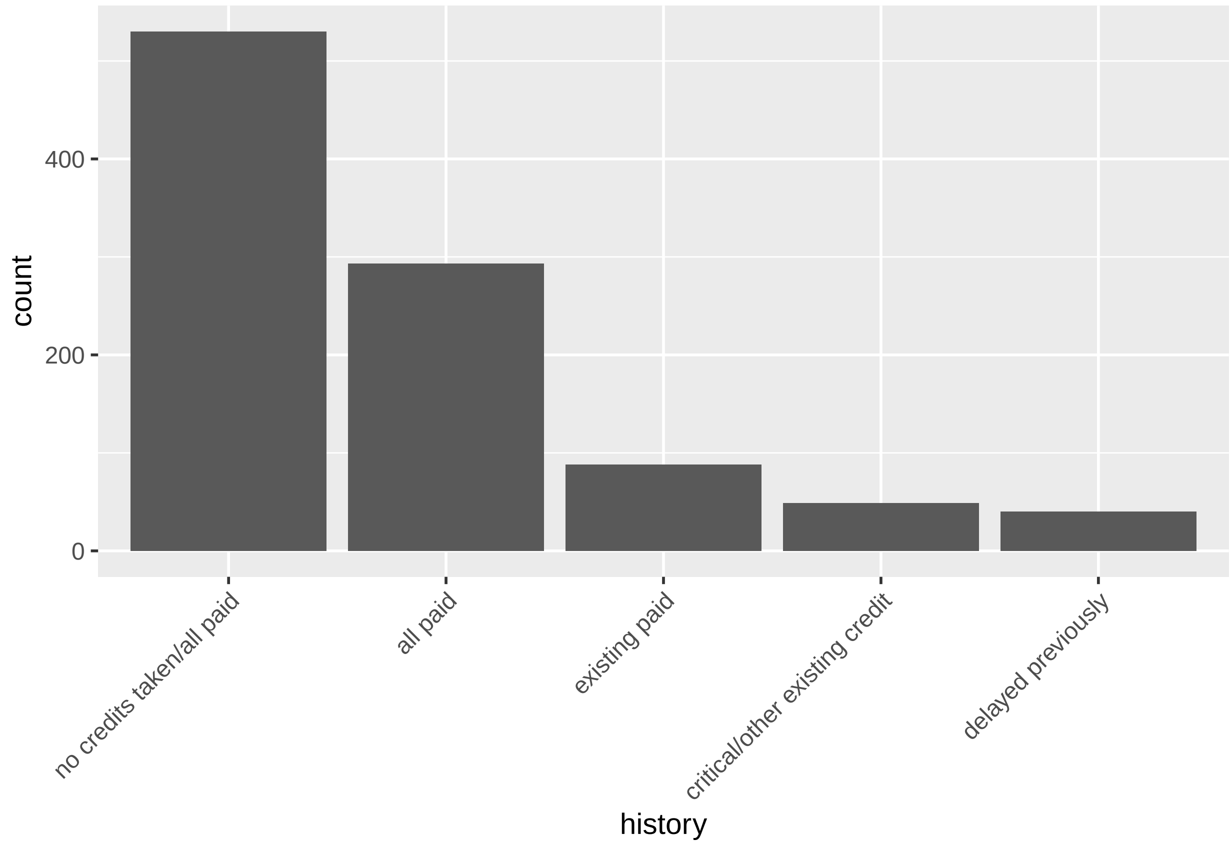
```
attach(df)
par(mfrow = c(2,2))
plot(sort(checking, decreasing = T))
plot(history)
plot(purpose)
plot(savings)
```



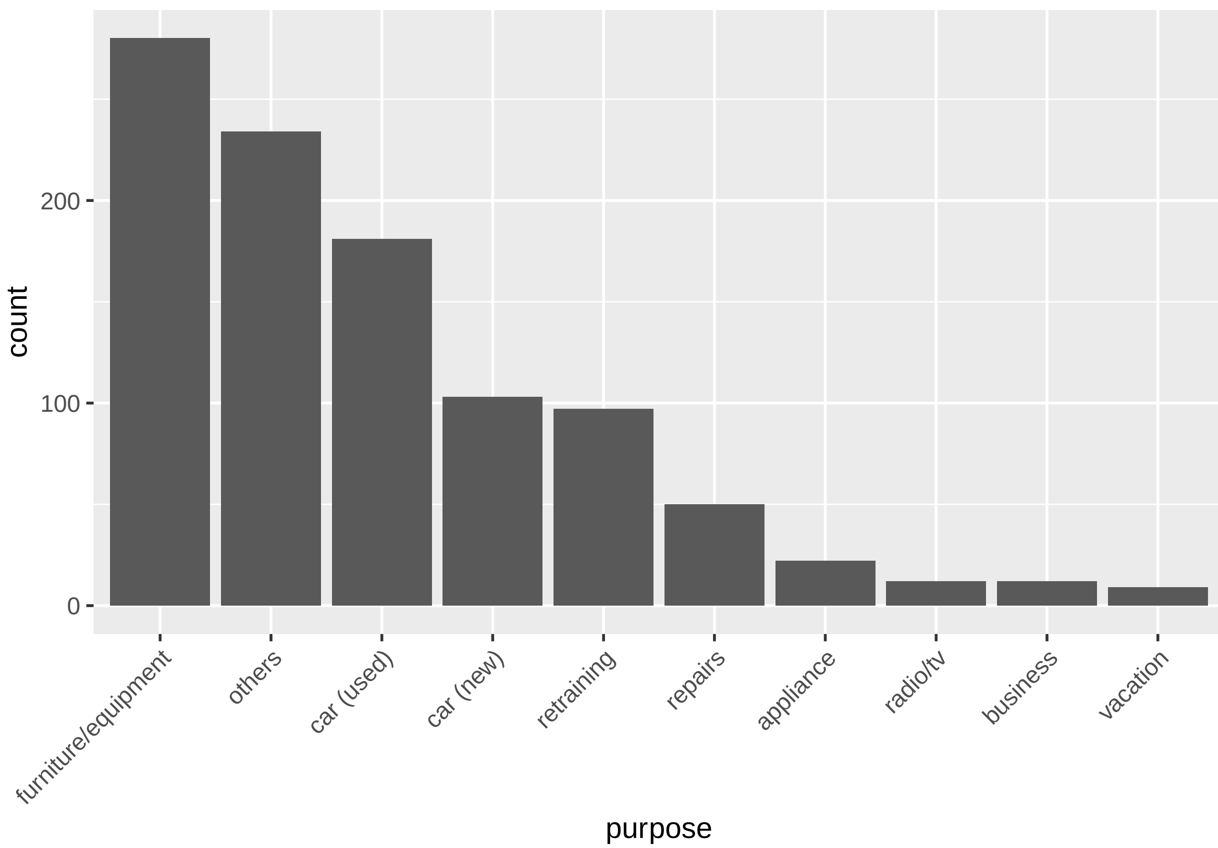

```r
library(forcats)
par(mfrow = c(2,2))
ggplot(mutate(df, checking =fct_infreq(checking))) + geom_bar(aes(x = checking))
```



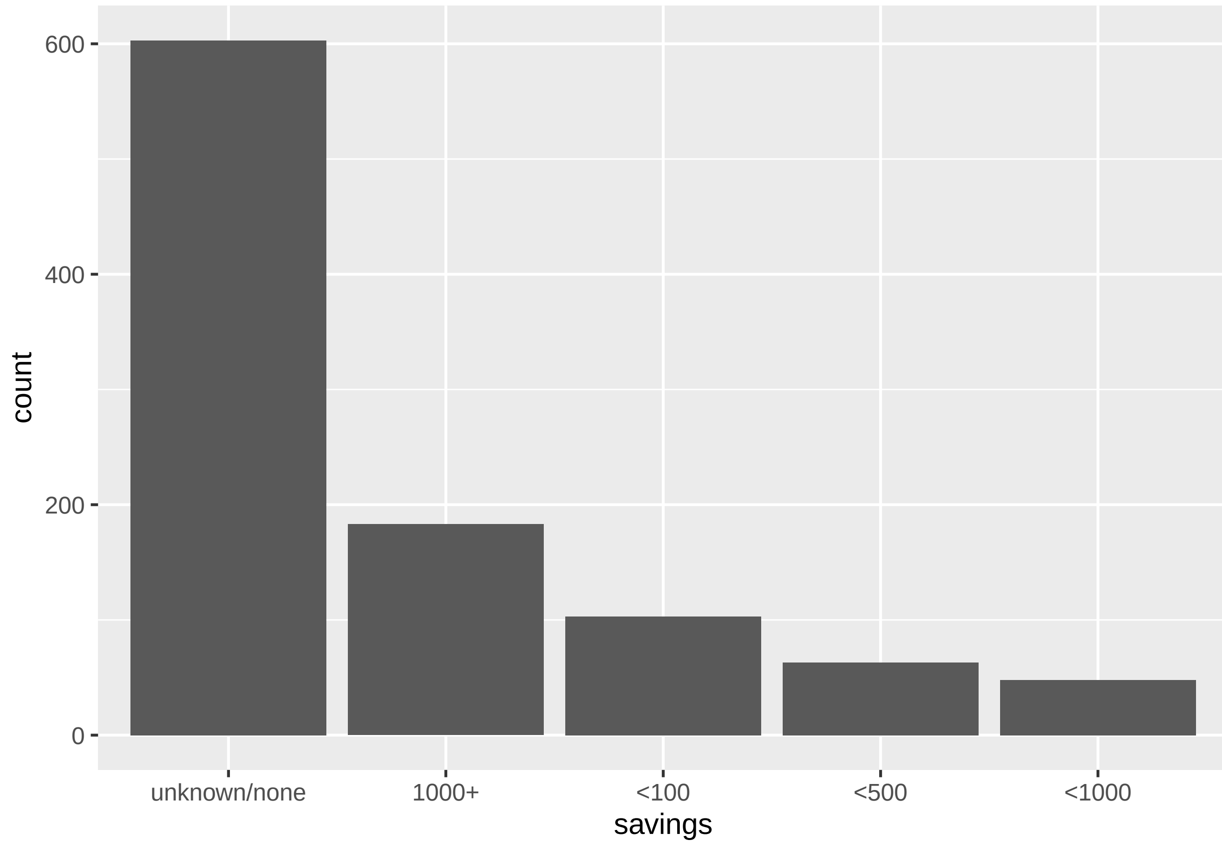

```
ggplot(mutate(df, history =fct_infreq(history))) + geom_bar(aes(x = history)) + theme(axis.text.x = eler
```



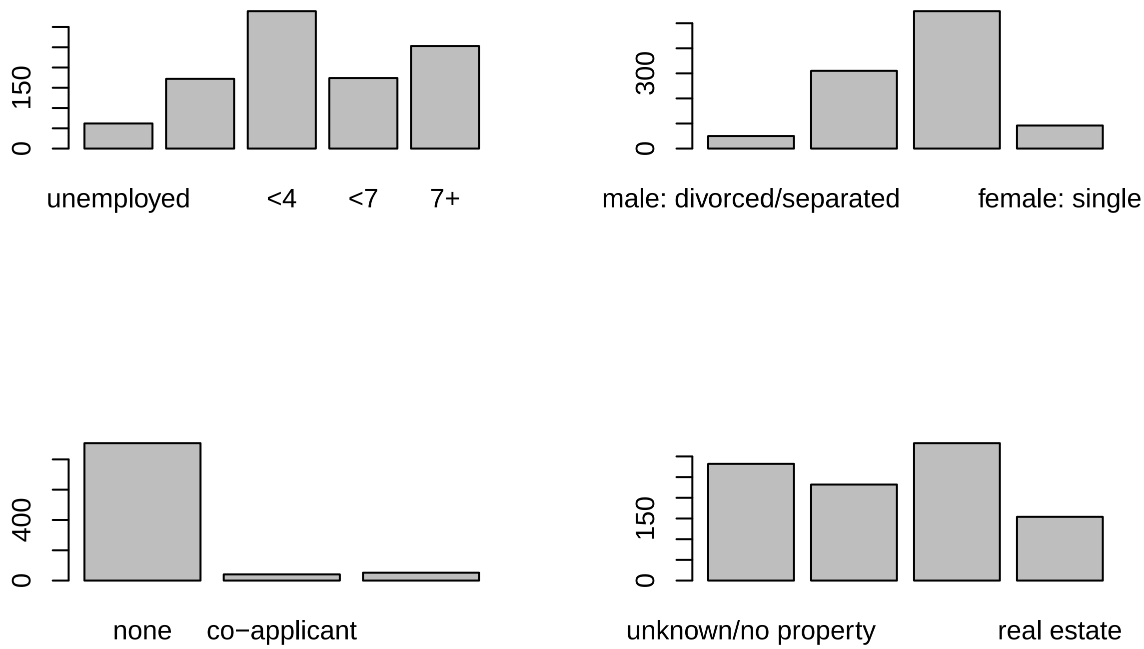

```
ggplot(mutate(df, purpose =fct_infreq(purpose))) + geom_bar(aes(x = purpose)) + theme(axis.text.x = elem
```




```
ggplot(mutate(df, savings =fct_infreq(savings))) + geom_bar(aes(x = savings))
```

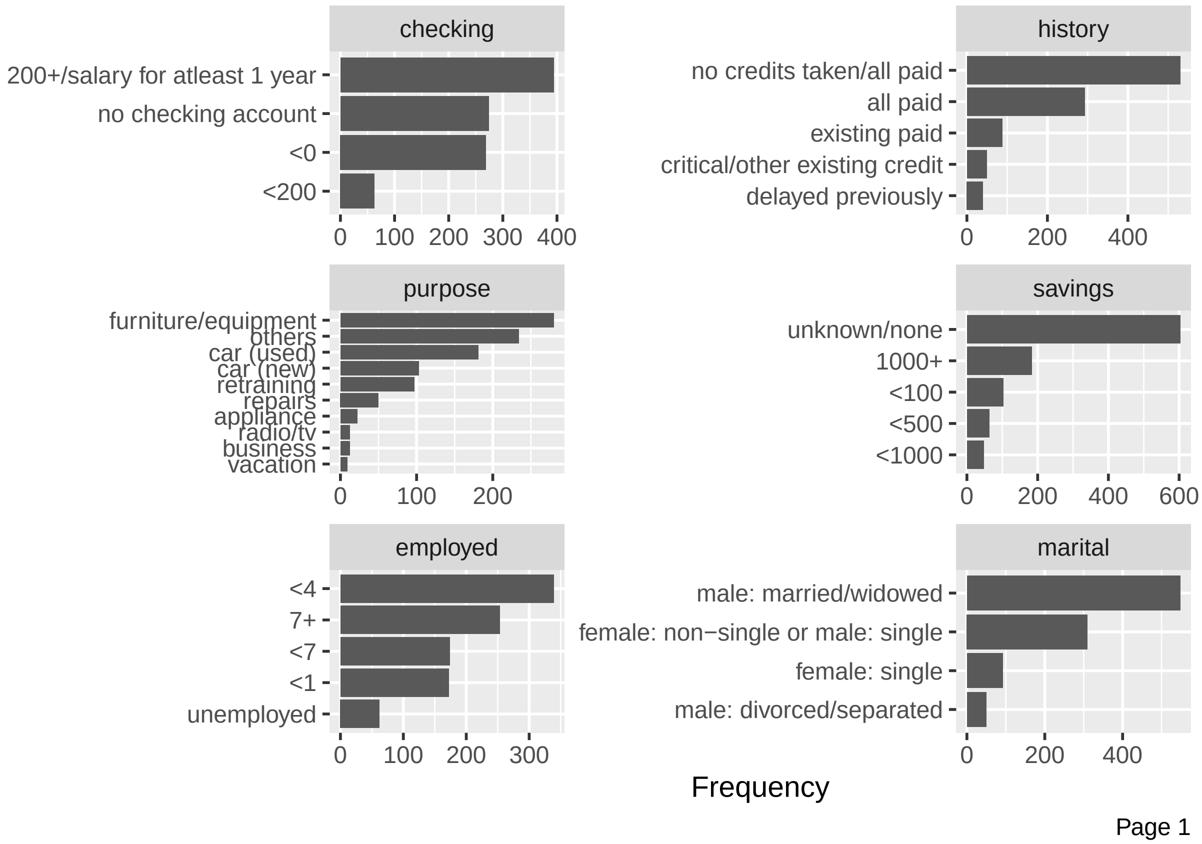



```
plot(employed)
plot(marital)
plot(coapp)
plot(property)
```

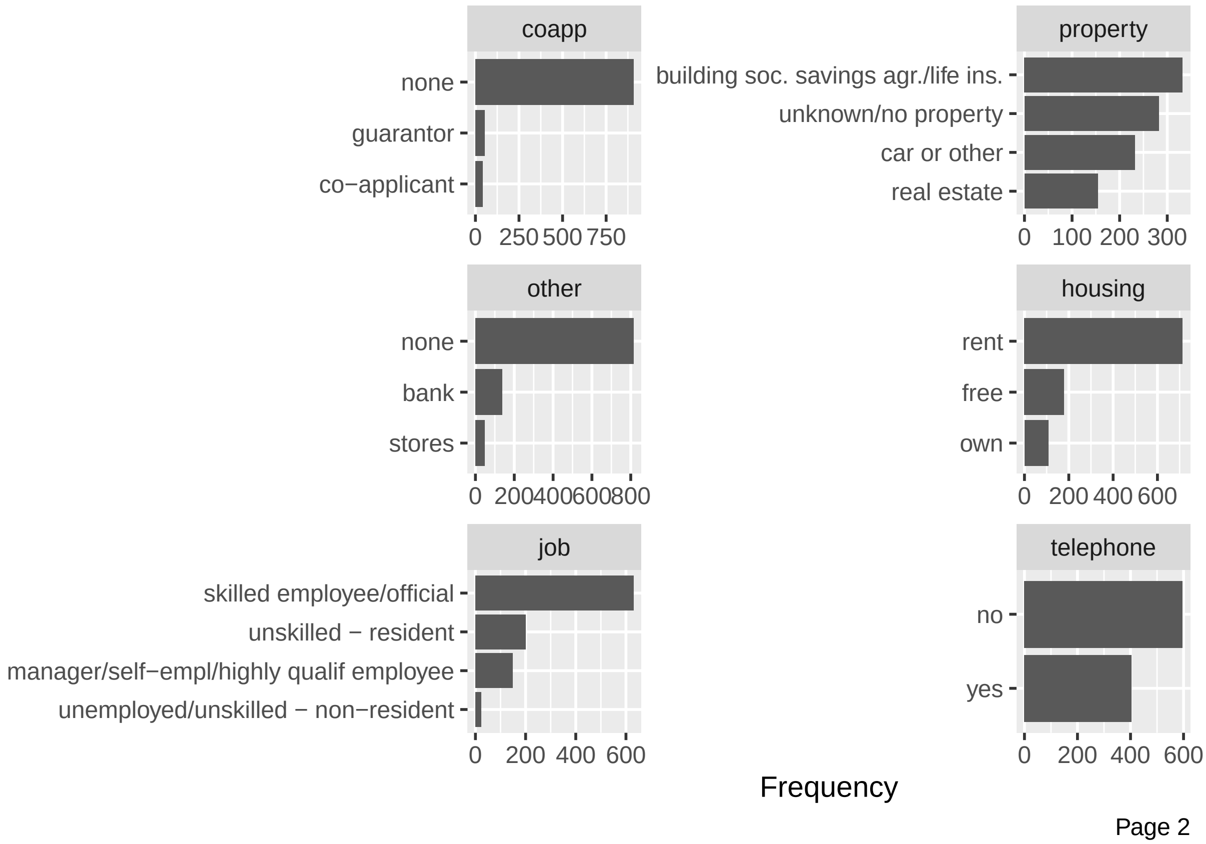

0
150
unemployed
<4
<7
7+
0
300
male: divorced/separated
female: single
0
400
none
co−applicant
0
150
unknown/no property
real estate

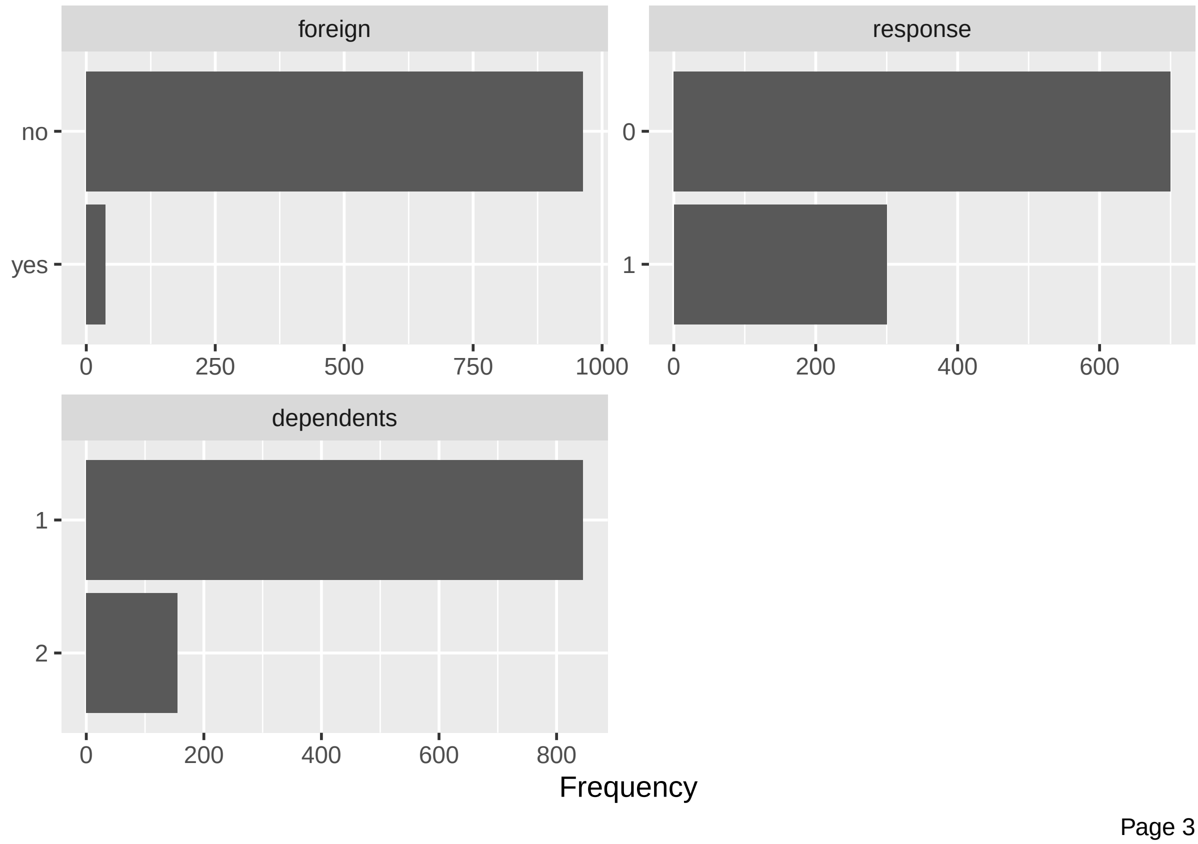
```
plot(other)
```


```r
plot_bar(df, ncol = 2)
```

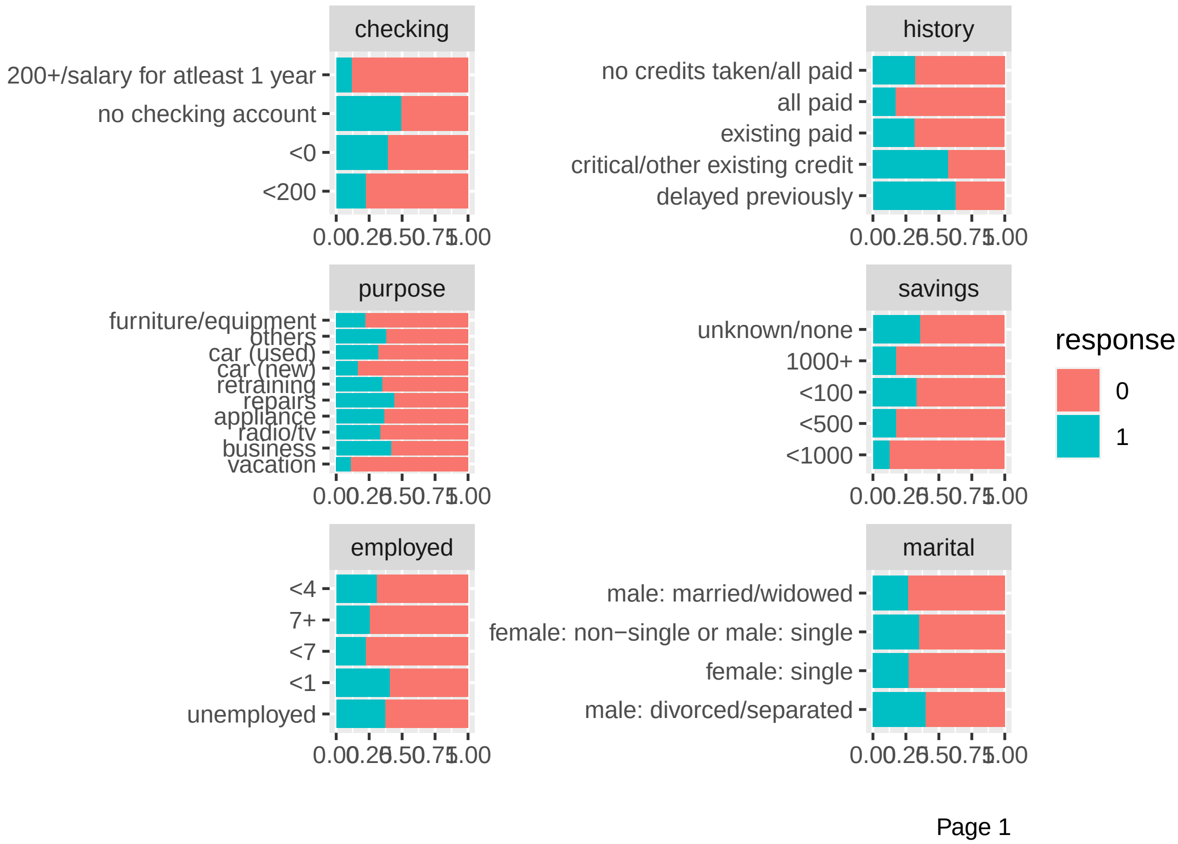


checking
200+/salary for atleast 1 year
no checking account
<0
<200
0
100
200
300
400
history
no credits taken/all paid
all paid
existing paid
critical/other existing credit
delayed previously
0
200
400
purpose
furniture/equipment
others
car (used)
car (new)
retraining
repairs
appliance
radio/tv
business
vacation
0
100
200
savings
unknown/none
1000+
<100
<500
<1000
0
200
400
600
employed
<4
7+
<7
<1
unemployed
0
100
200
300
marital
male: married/widowed
female: non−single or male: single
female: single
male: divorced/separated
0
200
400
Frequency
Page 1

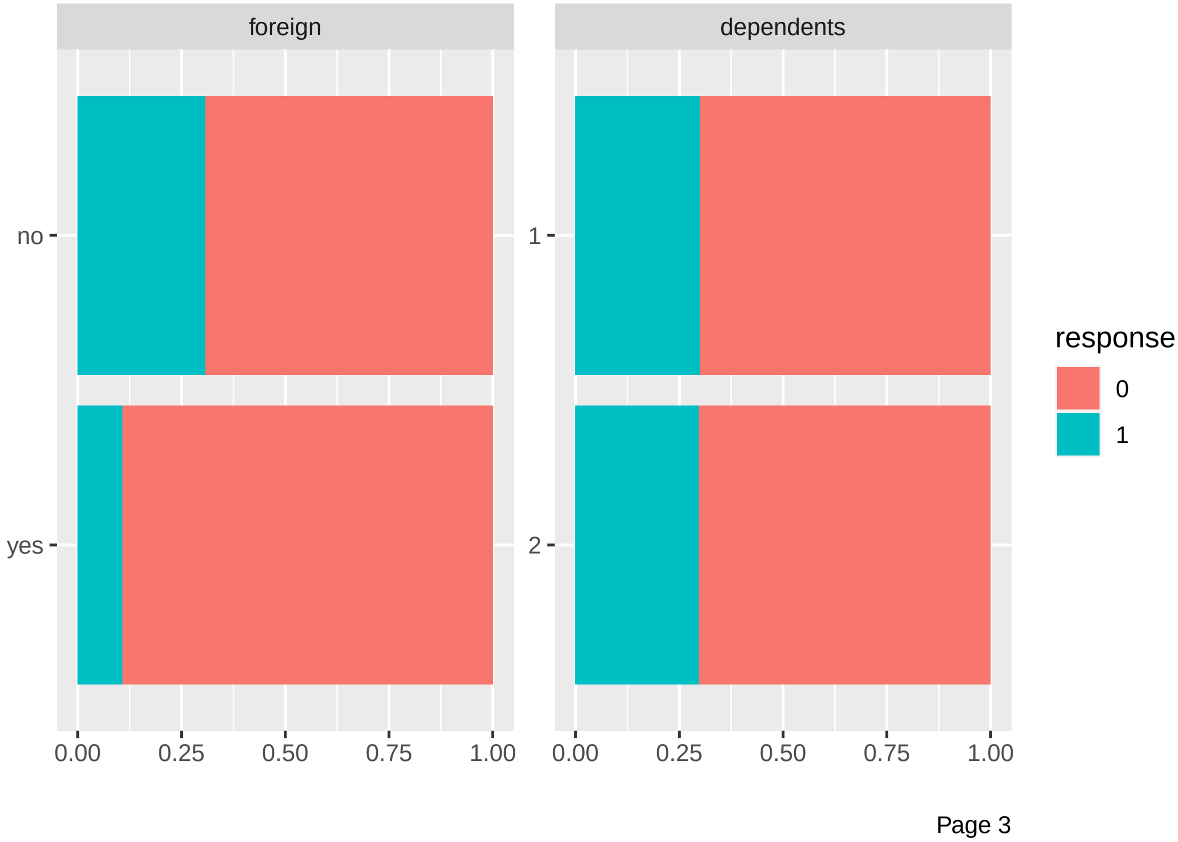

coapp
none
guarantor
co−applicant
0
250
500
750
property
building soc. savings agr./life ins.
unknown/no property
car or other
real estate
0
100
200
300
other
none
bank
stores
0
200
400
600
800
housing
rent
free
own
0
200
400
600
job
skilled employee/official
unskilled − resident
manager/self−empl/highly qualif employee
unemployed/unskilled − non−resident
0
200
400
600
telephone
no
yes
0
200
400
600
Frequency
Page 2



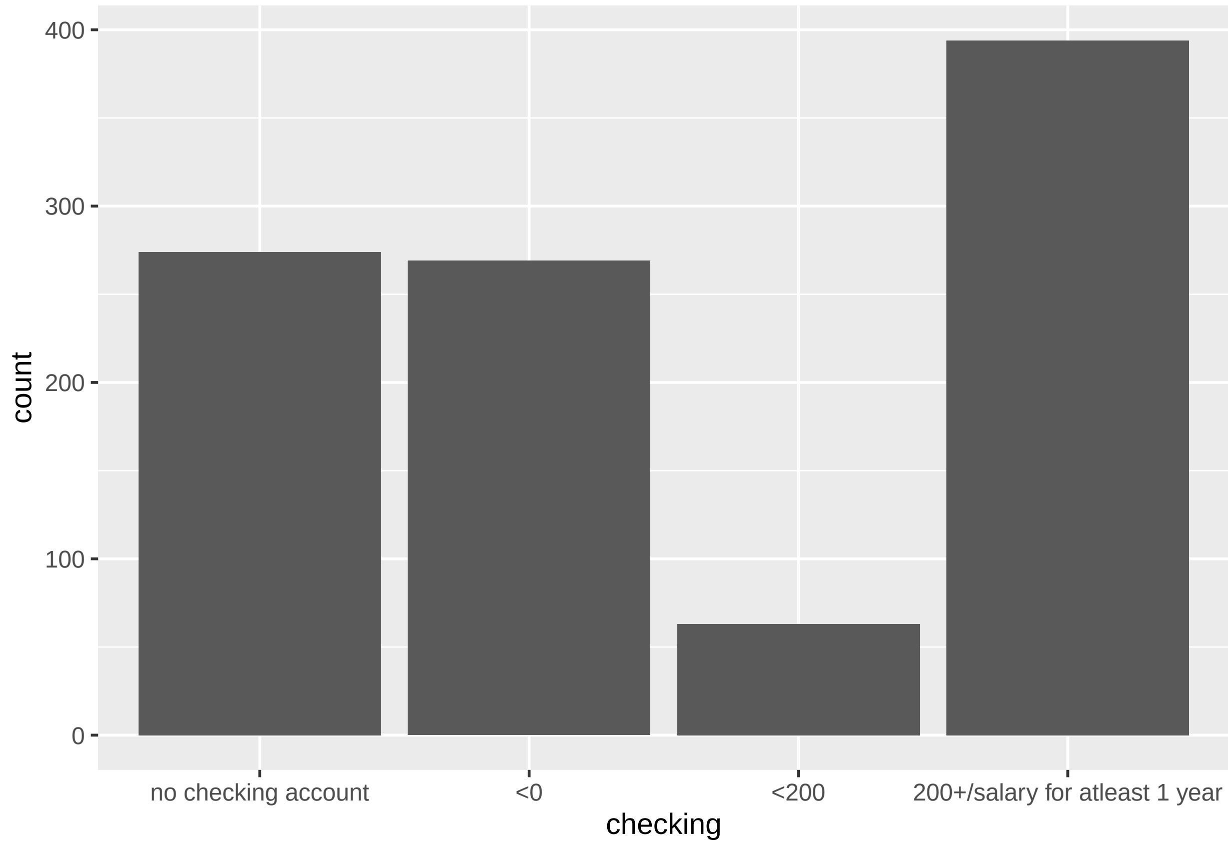

```r
plot_bar(df, by = "response", ncol = 2)
```


checking
200+/salary for atleast 1 year
no checking account
<0
<200
history
no credits taken/all paid
all paid
existing paid
critical/other existing credit
delayed previously
purpose
furniture/equipment
others
car (used)
car (new)
retraining
repairs
appliance
radio/tv
business
vacation
savings
unknown/none
1000+
<100
<500
<1000
employed
<4
7+
<7
<1
unemployed
marital
male: married/widowed
female: non−single or male: single
female: single
male: divorced/separated
0.00
0.25
0.50
0.75
1.00
response
0
1
Page 1

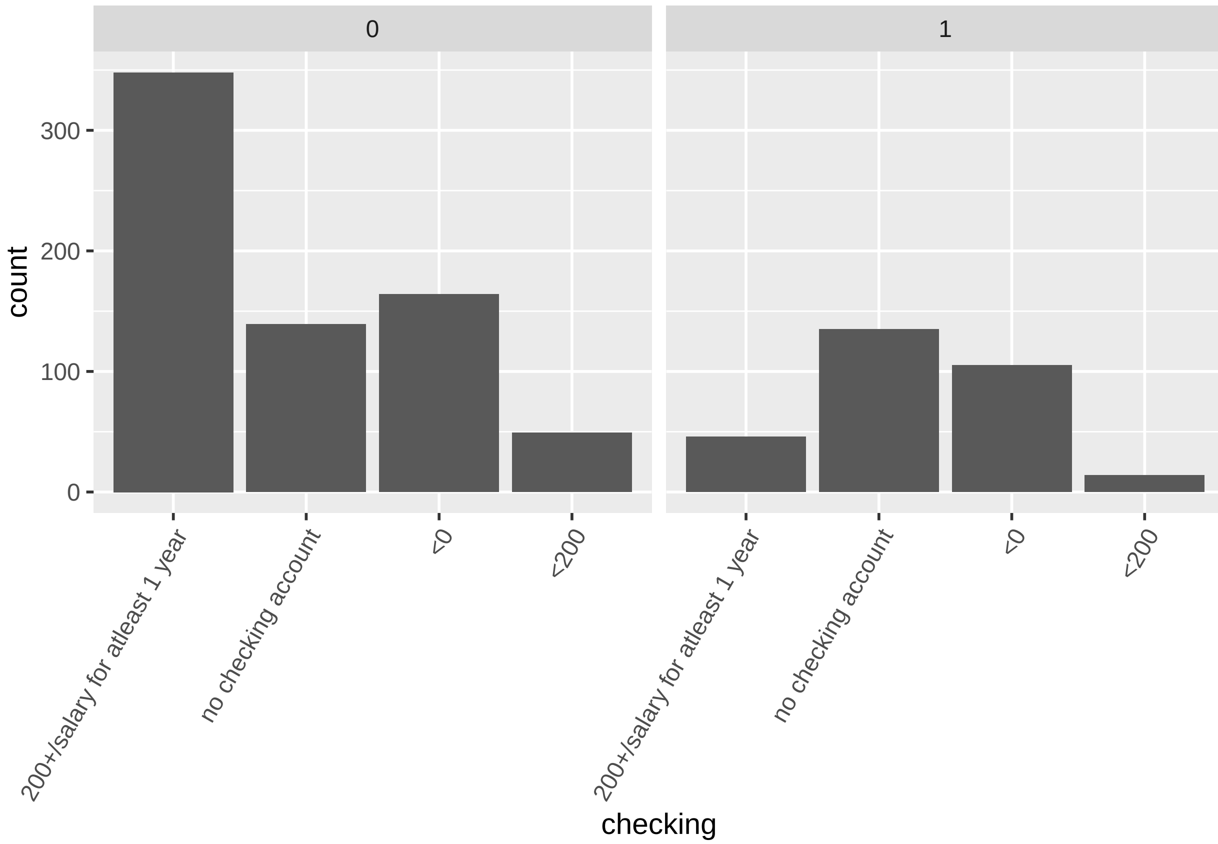

coapp
none
guarantor
co−applicant
property
building soc. savings agr./life ins.
unknown/no property
car or other
real estate
other
none
bank
stores
housing
rent
free
own
response
0
1
job
skilled employee/official
unskilled − resident
manager/self−empl/highly qualif employee
unemployed/unskilled − non−resident
elephon
no
yes
Page 2



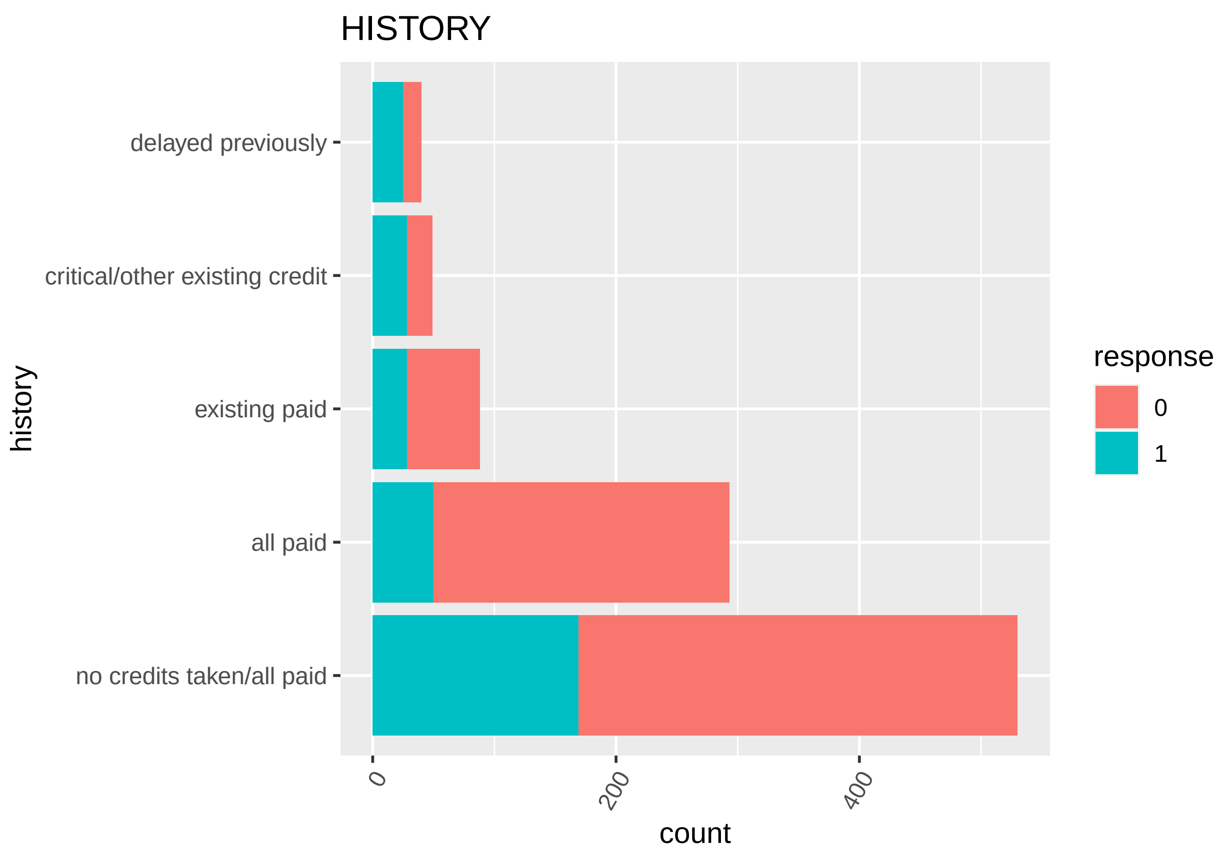

```
ggplot(df) + geom_bar(aes(x = checking))
```



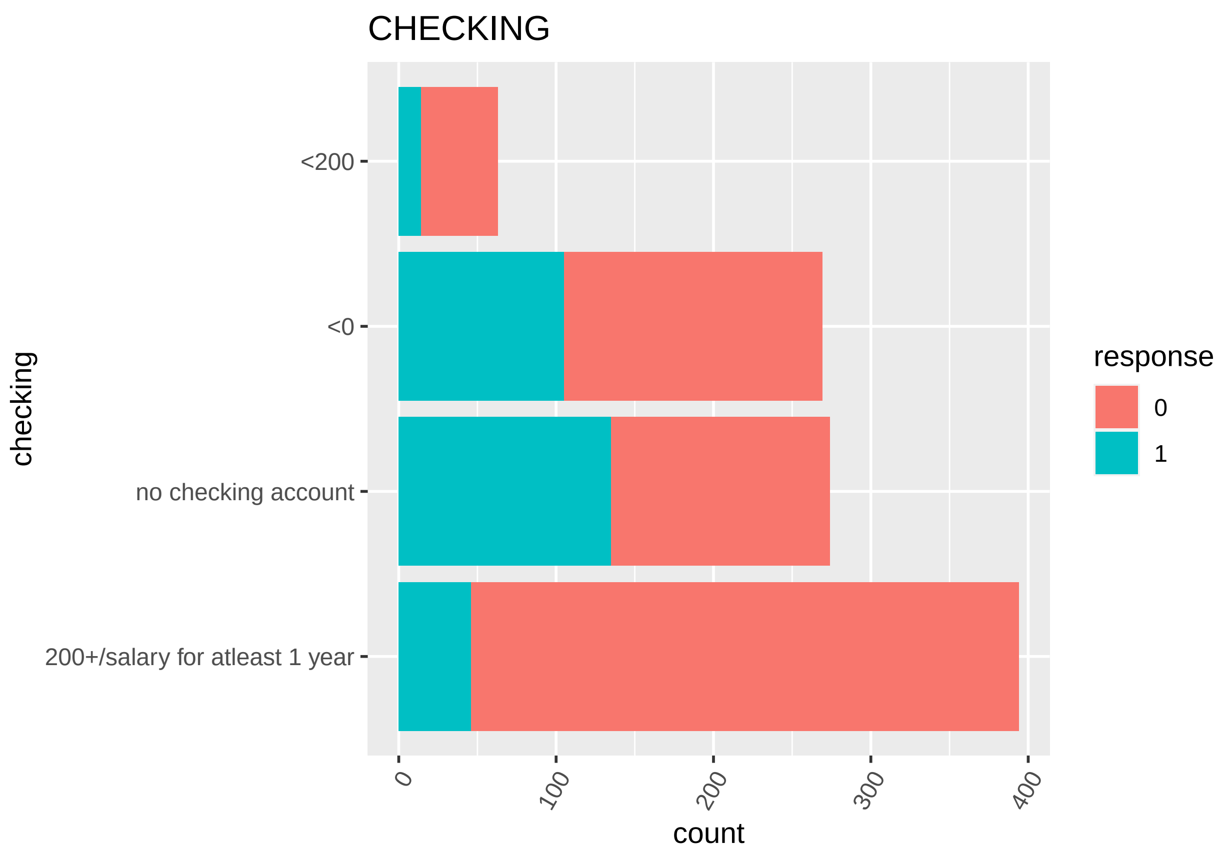

```
library(forcats)
ggplot(mutate(df, checking = fct_infreq(checking))) +
  geom_bar(aes(x = checking)) +
  facet_wrap(~response) +
  theme(axis.text.x = element_text(angle = 60, hjust = 1))
```



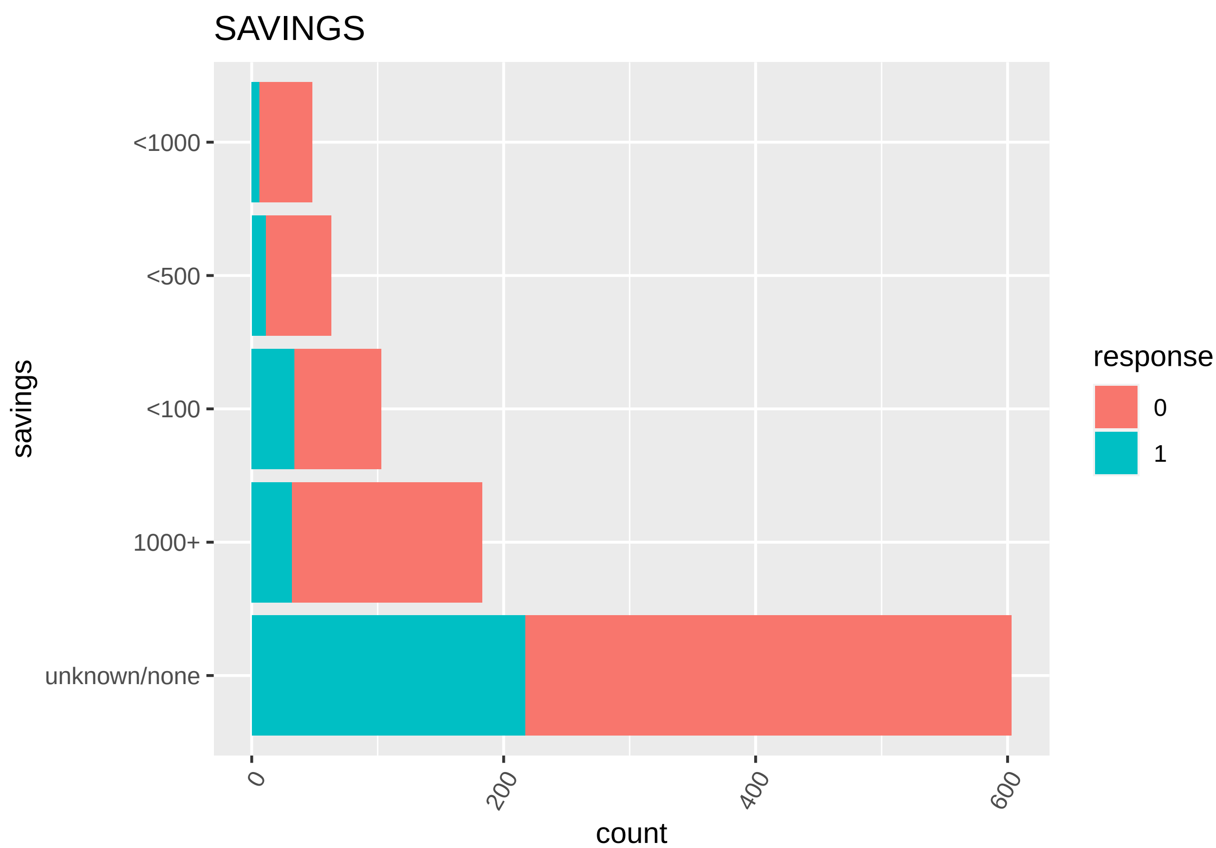

```
library(forcats)
ggplot(mutate(df, history = fct_infreq(history))) +
  geom_bar(aes(x = history, fill = response)) +
  theme(axis.text.x = element_text(angle = 60, hjust = 1)) +
  coord_flip()+
  ggtitle("HISTORY")
```



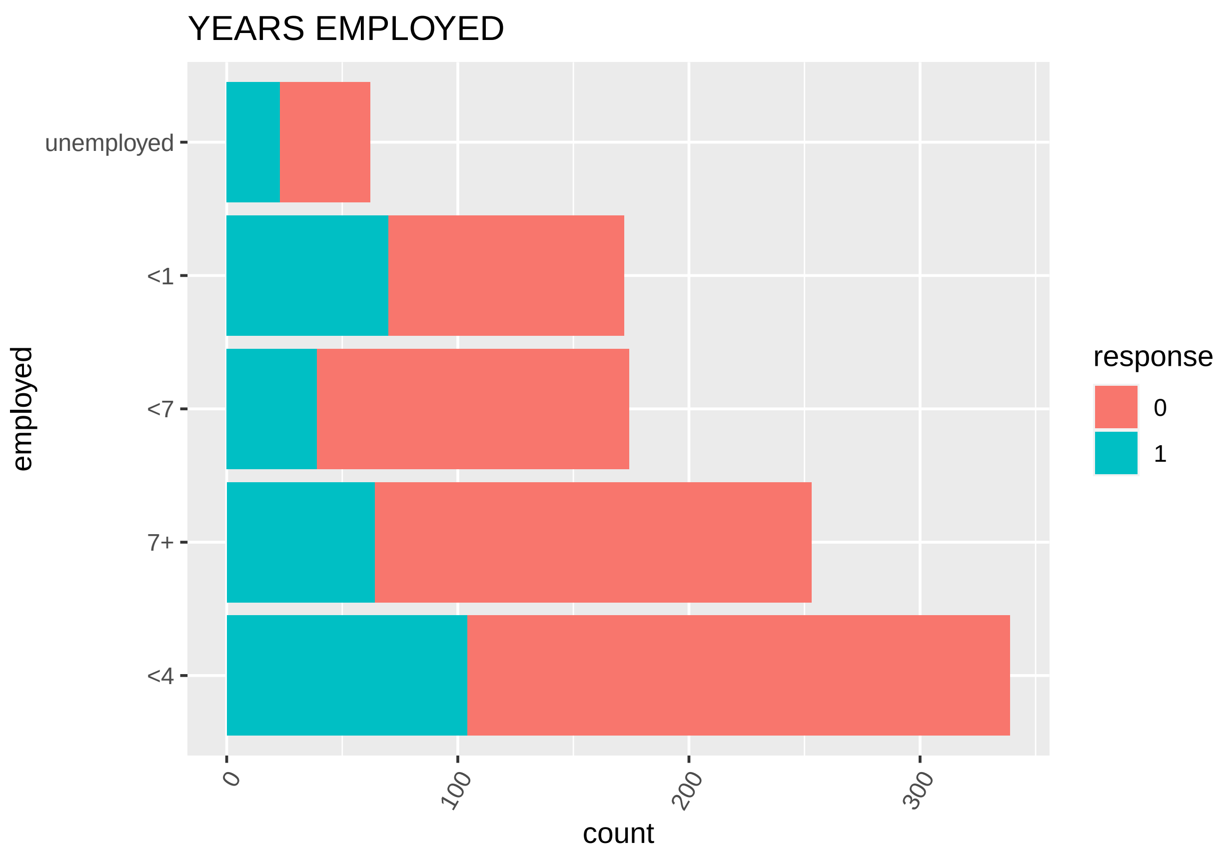

```r
library(forcats)
ggplot(mutate(df, checking = fct_infreq(checking))) +
  geom_bar(aes(x = checking, fill = response)) +
  theme(axis.text.x = element_text(angle = 60, hjust = 1)) +
  coord_flip()+
  ggtitle("CHECKING")
```




```
library(forcats)
ggplot(mutate(df, savings = fct_infreq(savings))) +
  geom_bar(aes(x = savings, fill = response)) +
  theme(axis.text.x = element_text(angle = 60, hjust = 1)) +
  coord_flip()+
  ggtitle("SAVINGS")
```




```
library(forcats)
ggplot(mutate(df, employed = fct_infreq(employed))) +
  geom_bar(aes(x = employed, fill = response)) +
  theme(axis.text.x = element_text(angle = 60, hjust = 1)) +
  coord_flip()+
  ggtitle("YEARS EMPLOYED")
```



SPLIT TEST AND TRAIN DATASET

```r
library(caret)
set.seed(2021)
index = createDataPartition(df$response, p=0.8, list = FALSE)
train = df[index,]
test = df[-index,]
```

PREDICTING RESPONSE

LOGIT1

```r
set.seed(2021)
logit1 = glm(response ~ ., data = train, family = binomial)
summary(logit1)
```

```
##
## Call:
## glm(formula = response ~ ., family = binomial, data = train)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3868  -0.6913  -0.3399   0.6157   2.7062
##
## Coefficients:
##                                              Estimate Std. Error z value
## (Intercept)                                -1.262e-01  1.311e+00  -0.096
## checking<0                                 -3.085e-01  2.482e-01  -1.243
## checking<200                               -5.225e-01  4.029e-01  -1.297
## checking200+/salary for atleast 1 year     -1.713e+00  2.704e-01  -6.334
## duration                                    3.580e-02  1.081e-02   3.314
## historycritical/other existing credit      -5.584e-02  6.311e-01  -0.088
## historyno credits taken/all paid           -8.950e-01  4.828e-01  -1.854
## historyexisting paid                       -1.162e+00  5.353e-01  -2.170
## historyall paid                            -1.911e+00  5.005e-01  -3.818
## purposecar (new)                           -1.762e+00  4.519e-01  -3.900
## purposecar (used)                          -1.055e+00  3.060e-01  -3.449
## purposefurniture/equipment                 -1.163e+00  2.879e-01  -4.038
## purposeradio/tv                            -3.117e-01  8.746e-01  -0.356
## purposeappliance                           -9.174e-01  6.926e-01  -1.324
## purposerepairs                              1.049e-01  4.598e-01   0.228
## purposevacation                            -2.498e+00  1.227e+00  -2.036
## purposeretraining                          -9.331e-01  3.909e-01  -2.387
## purposebusiness                            -7.921e-01  8.751e-01  -0.905
## amount                                      1.332e-04  5.184e-05   2.570
## savings<100                                -4.371e-01  3.305e-01  -1.323
## savings<500                                -2.491e-01  4.555e-01  -0.547
## savings<1000                               -1.446e+00  6.118e-01  -2.364
## savings1000+                               -1.135e+00  3.040e-01  -3.733
## employed<1                                  7.956e-02  5.011e-01   0.159
## employed<4                                 -3.006e-02  4.863e-01  -0.062
## employed<7                                 -7.070e-01  5.220e-01  -1.355
## employed7+                                 -8.572e-02  4.834e-01  -0.177
## installp                                    4.241e-01  1.022e-01   4.149
## maritalfemale: non-single or male: single  -2.408e-01  4.509e-01  -0.534
## maritalmale: married/widowed               -9.490e-01  4.487e-01  -2.115
## maritalfemale: single                      -4.269e-01  5.176e-01  -0.825
## coappco-applicant                           5.227e-01  4.826e-01   1.083
## coappguarantor                             -1.105e+00  4.695e-01  -2.353
## resident                                    9.606e-02  9.792e-02   0.981
## propertycar or other                        3.584e-01  2.965e-01   1.209
## propertybuilding soc. savings agr./life ins. 4.409e-02  2.714e-01   0.162
## propertyreal estate                         7.379e-01  4.979e-01   1.482
## age                                        -1.825e-02  1.075e-02  -1.698
## otherstores                                -4.035e-01  4.777e-01  -0.845
## othernone                                  -8.070e-01  2.773e-01  -2.910
## housingrent                                -2.980e-01  2.678e-01  -1.113
## housingown                                 -1.114e+00  5.627e-01  -1.980
## existcr                                     1.945e-01  2.195e-01   0.886
## jobunskilled - resident                     8.948e-01  8.743e-01   1.023
## jobskilled employee/official                7.125e-01  8.558e-01   0.833
## jobmanager/self-empl/highly qualif employee 8.896e-01  8.659e-01   1.027
## dependents                                  5.653e-01  2.905e-01   1.946
```

```
## telephoneyes                                   -3.896e-01  2.376e-01  -1.639
## foreignyes                                     -1.357e+00  6.956e-01  -1.951
##                                                Pr(>|z|)
## (Intercept)                                    0.923346
## checking<0                                     0.214003
## checking<200                                   0.194725
## checking200+/salary for atleast 1 year         2.39e-10 ***
## duration                                       0.000921 ***
## historycritical/other existing credit          0.929490
## historyno credits taken/all paid               0.063785 .
## historyexisting paid                           0.030003 *
## historyall paid                                0.000135 ***
## purposecar (new)                               9.64e-05 ***
## purposecar (used)                              0.000563 ***
## purposefurniture/equipment                     5.39e-05 ***
## purposeradio/tv                                0.721557
## purposeappliance                               0.185342
## purposerepairs                                 0.819601
## purposevacation                                0.041767 *
## purposeretraining                              0.016982 *
## purposebusiness                                0.365376
## amount                                         0.010163 *
## savings<100                                    0.185929
## savings<500                                    0.584534
## savings<1000                                   0.018099 *
## savings1000+                                   0.000189 ***
## employed<1                                     0.873850
## employed<4                                     0.950706
## employed<7                                     0.175560
## employed7+                                     0.859243
## installp                                       3.34e-05 ***
## maritalfemale: non-single or male: single      0.593309
## maritalmale: married/widowed                   0.034443 *
## maritalfemale: single                          0.409478
## coappco-applicant                              0.278760
## coappguarantor                                 0.018599 *
## resident                                       0.326586
## propertycar or other                           0.226690
## propertybuilding soc. savings agr./life ins. 0.870949
## propertyreal estate                            0.138349
## age                                            0.089555 .
## otherstores                                    0.398342
## othernone                                      0.003617 **
## housingrent                                    0.265801
## housingown                                     0.047697 *
## existcr                                        0.375596
## jobunskilled - resident                        0.306117
## jobskilled employee/official                   0.405120
## jobmanager/self-empl/highly qualif employee  0.304211
## dependents                                     0.051665 .
## telephoneyes                                   0.101134
## foreignyes                                     0.051001 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 977.38  on 799  degrees of freedom
## Residual deviance: 687.28  on 751  degrees of freedom
## AIC: 785.28
##
## Number of Fisher Scoring iterations: 5
```

```
#which predictors are significant and calculate model fit statistics
significant_if = summary(logit1)$coeff[-1,4]<.05
logit1.significant = names(significant_if)[significant_if==TRUE]

logit1.significant
```

```
##  [1] "checking200+/salary for atleast 1 year"
##  [2] "duration"
##  [3] "historyexisting paid"
##  [4] "historyall paid"
##  [5] "purposecar (new)"
##  [6] "purposecar (used)"
##  [7] "purposefurniture/equipment"
##  [8] "purposevacation"
##  [9] "purposeretraining"
## [10] "amount"
## [11] "savings<1000"
## [12] "savings1000+"
## [13] "installp"
## [14] "maritalmale: married/widowed"
## [15] "coappguarantor"
## [16] "othernone"
## [17] "housingown"
```

```
AIC = AIC(logit1)
BIC = BIC(logit1)
cbind(AIC, BIC)
```

```
##           AIC     BIC
## [1,] 785.2838 1014.83
```

```
#make predictions
library(caret)
test$PredProb.logit1 = predict.glm(logit1, newdata=test, type = 'response')
test$Pred.logit1 = ifelse(test$PredProb.logit1 >= .5,1,0)
caret::confusionMatrix(as.factor(test$Pred.logit1), as.factor(test$response))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 113  29
##          1  27  31
```

```
##
##                Accuracy : 0.72
##                  95% CI : (0.6523, 0.781)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 0.2972
##
##                   Kappa : 0.3269
##
##  Mcnemar's Test P-Value : 0.8937
##
##             Sensitivity : 0.8071
##             Specificity : 0.5167
##          Pos Pred Value : 0.7958
##          Neg Pred Value : 0.5345
##              Prevalence : 0.7000
##          Detection Rate : 0.5650
##    Detection Prevalence : 0.7100
##       Balanced Accuracy : 0.6619
##
##        'Positive' Class : 0
##
```

```
#calculate auc
library(ROCR)
library(pROC)
library(car)
pred1 = prediction(predict(logit1, test, type = "response"), test$response)
auc1 = round(as.numeric(performance(pred1, measure = "auc")@y.values), 3)
auc1
```

```
## [1] 0.748
```

```
library(car)
vif(logit1)
```

```
##                GVIF Df GVIF^(1/(2*Df))
## checking   1.447506  3        1.063580
## duration   1.929758  1        1.389157
## history    2.557322  4        1.124536
## purpose    3.448059  9        1.071187
## amount     2.501563  1        1.581633
## savings    1.483796  4        1.050562
## employed   2.351738  4        1.112817
## installp   1.405959  1        1.185731
## marital    1.691936  3        1.091601
## coapp      1.273336  2        1.062272
## resident   1.349097  1        1.161506
## property   4.363689  3        1.278328
## age        1.525414  1        1.235077
## other      1.324376  2        1.072761
## housing    3.862088  2        1.401863
## existcr    1.747702  1        1.322007
## job        2.478633  3        1.163328
```

```
## dependents 1.267822  1        1.125976
## telephone  1.484778  1        1.218515
## foreign    1.114868  1        1.055873
```

LOGIT2

```r
set.seed(2021)
logit2 = glm(response ~ checking + duration + history + purpose + amount + savings + installp + marital
summary(logit2)
```

```
##
## Call:
## glm(formula = response ~ checking + duration + history + purpose +
##     amount + savings + installp + marital + coapp + other + housing,
##     family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2542  -0.6814  -0.3682   0.7130   2.8244
##
## Coefficients:
##                                          Estimate Std. Error z value
## (Intercept)                             1.1156755  0.7443497   1.499
## checking<0                             -0.3330485  0.2379398  -1.400
## checking<200                           -0.6655531  0.3919848  -1.698
## checking200+/salary for atleast 1 year  -1.7428929  0.2621858  -6.648
## duration                                0.0350884  0.0101055   3.472
## historycritical/other existing credit  -0.2108572  0.6005717  -0.351
## historyno credits taken/all paid       -0.9919371  0.4623314  -2.146
## historyexisting paid                   -1.1520421  0.5262134  -2.189
## historyall paid                        -1.9273389  0.4901564  -3.932
## purposecar (new)                       -1.5499565  0.4273393  -3.627
## purposecar (used)                      -0.8791784  0.2875044  -3.058
## purposefurniture/equipment             -1.1233727  0.2749788  -4.085
## purposeradio/tv                        -0.4807177  0.8568491  -0.561
## purposeappliance                       -0.7972303  0.6703029  -1.189
## purposerepairs                          0.2296416  0.4495103   0.511
## purposevacation                        -2.3254044  1.2478288  -1.864
## purposeretraining                      -0.9748692  0.3762102  -2.591
## purposebusiness                        -0.8994886  0.8424164  -1.068
## amount                                  0.0001087  0.0000467   2.327
## savings<100                            -0.3975650  0.3143590  -1.265
## savings<500                            -0.3556678  0.4435836  -0.802
## savings<1000                           -1.4251200  0.5988878  -2.380
## savings1000+                           -1.1336720  0.2928607  -3.871
## installp                                0.3923391  0.0964465   4.068
## maritalfemale: non-single or male: single -0.1020225  0.4321289  -0.236
## maritalmale: married/widowed           -0.7388057  0.4275840  -1.728
## maritalfemale: single                  -0.3253385  0.4944843  -0.658
## coappco-applicant                       0.5724214  0.4593386   1.246
## coappguarantor                         -1.1308211  0.4576789  -2.471
```

```
## otherstores                              -0.2910166  0.4642153  -0.627
## othernone                                -0.7628315  0.2692630  -2.833
## housingrent                              -0.3735332  0.2494122  -1.498
## housingown                               -0.6261181  0.3790408  -1.652
##                                          Pr(>|z|)
## (Intercept)                              0.133910
## checking<0                               0.161598
## checking<200                             0.089526 .
## checking200+/salary for atleast 1 year   2.98e-11 ***
## duration                                 0.000516 ***
## historycritical/other existing credit    0.725518
## historyno credits taken/all paid         0.031912 *
## historyexisting paid                     0.028575 *
## historyall paid                          8.42e-05 ***
## purposecar (new)                         0.000287 ***
## purposecar (used)                        0.002228 **
## purposefurniture/equipment               4.40e-05 ***
## purposeradio/tv                          0.574777
## purposeappliance                         0.234299
## purposerepairs                           0.609442
## purposevacation                          0.062383 .
## purposeretraining                        0.009562 **
## purposebusiness                          0.285634
## amount                                   0.019977 *
## savings<100                              0.205984
## savings<500                              0.422665
## savings<1000                             0.017331 *
## savings1000+                             0.000108 ***
## installp                                 4.74e-05 ***
## maritalfemale: non-single or male: single 0.813361
## maritalmale: married/widowed             0.084013 .
## maritalfemale: single                    0.510580
## coappco-applicant                        0.212696
## coappguarantor                           0.013482 *
## otherstores                              0.530725
## othernone                                0.004611 **
## housingrent                              0.134223
## housingown                               0.098565 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 977.38  on 799  degrees of freedom
## Residual deviance: 712.89  on 767  degrees of freedom
## AIC: 778.89
##
## Number of Fisher Scoring iterations: 5
```

```r
#which predictors are significant and calculate model fit statistics
significant_if = summary(logit2)$coeff[-1,4]<.05
logit2.significant = names(significant_if)[significant_if==TRUE]

logit2.significant
```

```
##  [1] "checking200+/salary for atleast 1 year"
##  [2] "duration"
##  [3] "historyno credits taken/all paid"
##  [4] "historyexisting paid"
##  [5] "historyall paid"
##  [6] "purposecar (new)"
##  [7] "purposecar (used)"
##  [8] "purposefurniture/equipment"
##  [9] "purposeretraining"
## [10] "amount"
## [11] "savings<1000"
## [12] "savings1000+"
## [13] "installp"
## [14] "coappguarantor"
## [15] "othernone"
```

```r
AIC = AIC(logit2)
BIC = BIC(logit2)
cbind(AIC, BIC)
```

```
##           AIC      BIC
## [1,] 778.8892 933.4814
```

```r
#make predictions
library(caret)
test$PredProb.logit2 = predict.glm(logit2, newdata=test, type = 'response')
test$Pred.logit2 = ifelse(test$PredProb.logit2 >= .5,1,0)
caret::confusionMatrix(as.factor(test$Pred.logit2), as.factor(test$response))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 114  32
##          1  26  28
##
##                Accuracy : 0.71
##                  95% CI : (0.6418, 0.7718)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 0.4123
##
##                   Kappa : 0.2892
##
##  Mcnemar's Test P-Value : 0.5115
##
##             Sensitivity : 0.8143
##             Specificity : 0.4667
##          Pos Pred Value : 0.7808
##          Neg Pred Value : 0.5185
##              Prevalence : 0.7000
##          Detection Rate : 0.5700
##    Detection Prevalence : 0.7300
##       Balanced Accuracy : 0.6405
```

```
##
##        'Positive' Class : 0
##
```

```
#calculate auc
library(ROCR)
library(pROC)
library(car)
pred2 = prediction(predict(logit2, test, type = "response"), test$response)
auc2 = round(as.numeric(performance(pred2, measure = "auc")@y.values), 3)
auc2
```

```
## [1] 0.753
```

```
vif(logit2)
```

```
##              GVIF Df GVIF^(1/(2*Df))
## checking 1.325307  3        1.048060
## duration 1.753516  1        1.324204
## history  1.487246  4        1.050867
## purpose  2.269301  9        1.046578
## amount   2.088649  1        1.445216
## savings  1.297461  4        1.033087
## installp 1.291920  1        1.136627
## marital  1.332339  3        1.048985
## coapp    1.174191  2        1.040962
## other    1.227912  2        1.052669
## housing  1.340525  2        1.076016
```

LOGIT3

```
set.seed(2021)
logit3 = glm(response ~ checking + duration + history + purpose + amount + savings + installp + marital
summary(logit3)
```

```
##
## Call:
## glm(formula = response ~ checking + duration + history + purpose +
##     amount + savings + installp + marital + coapp + other, family = binomial,
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3550  -0.6942  -0.3726   0.7387   2.7431
##
## Coefficients:
##                                            Estimate Std. Error z value
## (Intercept)                               8.569e-01  7.214e-01   1.188
## checking<0                               -3.531e-01  2.366e-01  -1.493
```

```
## checking<200                             -6.937e-01  3.897e-01  -1.780
## checking200+/salary for atleast 1 year    -1.764e+00  2.596e-01  -6.797
## duration                                   3.352e-02  1.004e-02   3.337
## historycritical/other existing credit     -2.443e-01  6.010e-01  -0.407
## historyno credits taken/all paid          -1.023e+00  4.597e-01  -2.226
## historyexisting paid                      -1.169e+00  5.244e-01  -2.228
## historyall paid                           -1.955e+00  4.871e-01  -4.014
## purposecar (new)                          -1.586e+00  4.246e-01  -3.734
## purposecar (used)                         -8.587e-01  2.861e-01  -3.001
## purposefurniture/equipment                -1.110e+00  2.722e-01  -4.076
## purposeradio/tv                           -4.685e-01  8.559e-01  -0.547
## purposeappliance                          -8.040e-01  6.613e-01  -1.216
## purposerepairs                             2.002e-01  4.440e-01   0.451
## purposevacation                           -2.293e+00  1.252e+00  -1.831
## purposeretraining                         -9.456e-01  3.713e-01  -2.547
## purposebusiness                           -1.017e+00  8.327e-01  -1.221
## amount                                     1.070e-04  4.663e-05   2.295
## savings<100                               -3.560e-01  3.118e-01  -1.142
## savings<500                               -3.663e-01  4.441e-01  -0.825
## savings<1000                              -1.408e+00  5.996e-01  -2.348
## savings1000+                              -1.110e+00  2.913e-01  -3.810
## installp                                   3.845e-01  9.595e-02   4.007
## maritalfemale: non-single or male: single -6.258e-02  4.269e-01  -0.147
## maritalmale: married/widowed              -7.841e-01  4.226e-01  -1.855
## maritalfemale: single                     -3.033e-01  4.905e-01  -0.618
## coappco-applicant                          6.289e-01  4.580e-01   1.373
## coappguarantor                            -1.058e+00  4.505e-01  -2.350
## otherstores                               -3.175e-01  4.623e-01  -0.687
## othernone                                 -7.355e-01  2.680e-01  -2.744
##                                           Pr(>|z|)
## (Intercept)                               0.234876
## checking<0                                0.135484
## checking<200                              0.075077 .
## checking200+/salary for atleast 1 year    1.07e-11 ***
## duration                                  0.000846 ***
## historycritical/other existing credit     0.684322
## historyno credits taken/all paid          0.026039 *
## historyexisting paid                      0.025857 *
## historyall paid                           5.97e-05 ***
## purposecar (new)                          0.000188 ***
## purposecar (used)                         0.002691 **
## purposefurniture/equipment                4.57e-05 ***
## purposeradio/tv                           0.584108
## purposeappliance                          0.224043
## purposerepairs                            0.652099
## purposevacation                           0.067086 .
## purposeretraining                         0.010875 *
## purposebusiness                           0.221955
## amount                                    0.021710 *
## savings<100                               0.253494
## savings<500                               0.409487
## savings<1000                              0.018876 *
## savings1000+                              0.000139 ***
## installp                                  6.14e-05 ***
```

```
## maritalfemale: non-single or male: single 0.883452
## maritalmale: married/widowed             0.063566 .
## maritalfemale: single                    0.536369
## coappco-applicant                        0.169706
## coappguarantor                           0.018790 *
## otherstores                              0.492252
## othernone                                0.006072 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 977.38  on 799  degrees of freedom
## Residual deviance: 716.18  on 769  degrees of freedom
## AIC: 778.18
##
## Number of Fisher Scoring iterations: 5
```

```
#which predictors are significant and calculate model fit statistics
significant_if = summary(logit3)$coeff[-1,4]<.05
logit3.significant = names(significant_if)[significant_if==TRUE]

logit3.significant
```

```
##  [1] "checking200+/salary for atleast 1 year"
##  [2] "duration"
##  [3] "historyno credits taken/all paid"
##  [4] "historyexisting paid"
##  [5] "historyall paid"
##  [6] "purposecar (new)"
##  [7] "purposecar (used)"
##  [8] "purposefurniture/equipment"
##  [9] "purposeretraining"
## [10] "amount"
## [11] "savings<1000"
## [12] "savings1000+"
## [13] "installp"
## [14] "coappguarantor"
## [15] "othernone"
```

```
AIC = AIC(logit3)
BIC = BIC(logit3)
cbind(AIC, BIC)
```

```
##           AIC      BIC
## [1,] 778.1772 923.4001
```

```
#make predictions
library(caret)
test$PredProb.logit3 = predict.glm(logit3, newdata=test, type = 'response')
test$Pred.logit3 = ifelse(test$PredProb.logit3 >= .5,1,0)
caret::confusionMatrix(as.factor(test$Pred.logit3), as.factor(test$response))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 113  32
##          1  27  28
##
##                Accuracy : 0.705
##                  95% CI : (0.6366, 0.7672)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 0.4733
##
##                   Kappa : 0.2805
##
##  Mcnemar's Test P-Value : 0.6025
##
##             Sensitivity : 0.8071
##             Specificity : 0.4667
##          Pos Pred Value : 0.7793
##          Neg Pred Value : 0.5091
##              Prevalence : 0.7000
##          Detection Rate : 0.5650
##    Detection Prevalence : 0.7250
##       Balanced Accuracy : 0.6369
##
##        'Positive' Class : 0
##
```

```r
#calculate auc
library(ROCR)
library(pROC)
library(car)
pred3 = prediction(predict(logit3, test, type = "response"), test$response)
auc3 = round(as.numeric(performance(pred3, measure = "auc")@y.values), 3)
auc3
```

```
## [1] 0.755
```

```r
library(car)
vif(logit3)
```

```
##              GVIF Df GVIF^(1/(2*Df))
## checking 1.297581  3        1.044373
## duration 1.735634  1        1.317435
## history  1.474880  4        1.049771
## purpose  2.033195  9        1.040210
## amount   2.091170  1        1.446088
## savings  1.271714  4        1.030502
## installp 1.289743  1        1.135669
## marital  1.264047  3        1.039826
## coapp    1.162123  2        1.038276
## other    1.200007  2        1.046637
```

```r
odds_ratio = exp(logit3$coefficients)
round(odds_ratio, 3)
```

```
##                                (Intercept) 
##                                      2.356 
##                                 checking<0 
##                                      0.702 
##                               checking<200 
##                                      0.500 
##     checking200+/salary for atleast 1 year 
##                                      0.171 
##                                   duration 
##                                      1.034 
##      historycritical/other existing credit 
##                                      0.783 
##           historyno credits taken/all paid 
##                                      0.359 
##                        historyexisting paid 
##                                      0.311 
##                            historyall paid 
##                                      0.142 
##                           purposecar (new) 
##                                      0.205 
##                          purposecar (used) 
##                                      0.424 
##                purposefurniture/equipment 
##                                      0.330 
##                            purposeradio/tv 
##                                      0.626 
##                           purposeappliance 
##                                      0.448 
##                             purposerepairs 
##                                      1.222 
##                            purposevacation 
##                                      0.101 
##                          purposeretraining 
##                                      0.388 
##                            purposebusiness 
##                                      0.362 
##                                     amount 
##                                      1.000 
##                                savings<100 
##                                      0.700 
##                                savings<500 
##                                      0.693 
##                               savings<1000 
##                                      0.245 
##                               savings1000+ 
##                                      0.330 
##                                   installp 
##                                      1.469 
## maritalfemale: non-single or male: single 
##                                      0.939
```

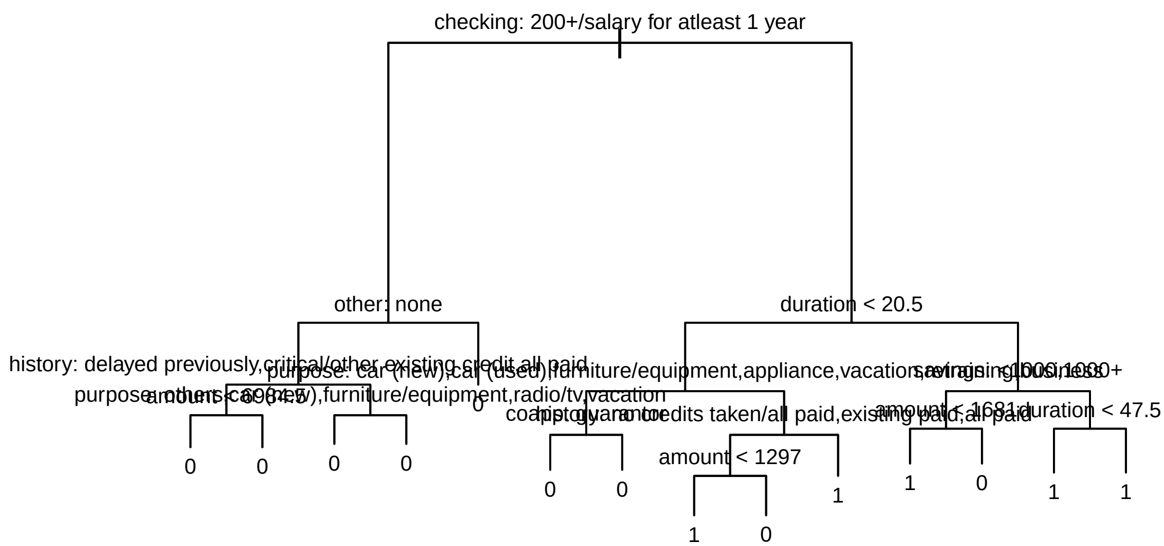
```
##              maritalmale: married/widowed
##                                     0.457
##                     maritalfemale: single
##                                     0.738
##                         coappco-applicant
##                                     1.875
##                            coappguarantor
##                                     0.347
##                               otherstores
##                                     0.728
##                                 othernone
##                                     0.479
```

DT1

```r
library(tree)
set.seed(2021)
DT1 = tree(response ~ . , train)
summary(DT1)
```

```
##
## Classification tree:
## tree(formula = response ~ ., data = train)
## Variables actually used in tree construction:
## [1] "checking" "other"    "history"  "amount"   "purpose"  "duration" "coapp"
## [8] "savings"
## Number of terminal nodes:  14
## Residual mean deviance:  0.8952 = 703.6 / 786
## Misclassification error rate: 0.2225 = 178 / 800
```

```r
plot(DT1)
text(DT1, pretty = 0, cex = 0.7)
```




```r
test$DT1.pred = predict(DT1, test, type = 'class')
caret::confusionMatrix(test$DT1.pred, test$response)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 109  31
##          1  31  29
##
##                Accuracy : 0.69
##                  95% CI : (0.6209, 0.7533)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 0.6533
##
##                   Kappa : 0.2619
##
##  Mcnemar's Test P-Value : 1.0000
##
##             Sensitivity : 0.7786
##             Specificity : 0.4833
##          Pos Pred Value : 0.7786
##          Neg Pred Value : 0.4833
##              Prevalence : 0.7000
##          Detection Rate : 0.5450
```

```
##    Detection Prevalence : 0.7000
##       Balanced Accuracy : 0.6310
##
##        'Positive' Class : 0
##
```

DT1_PRUNED

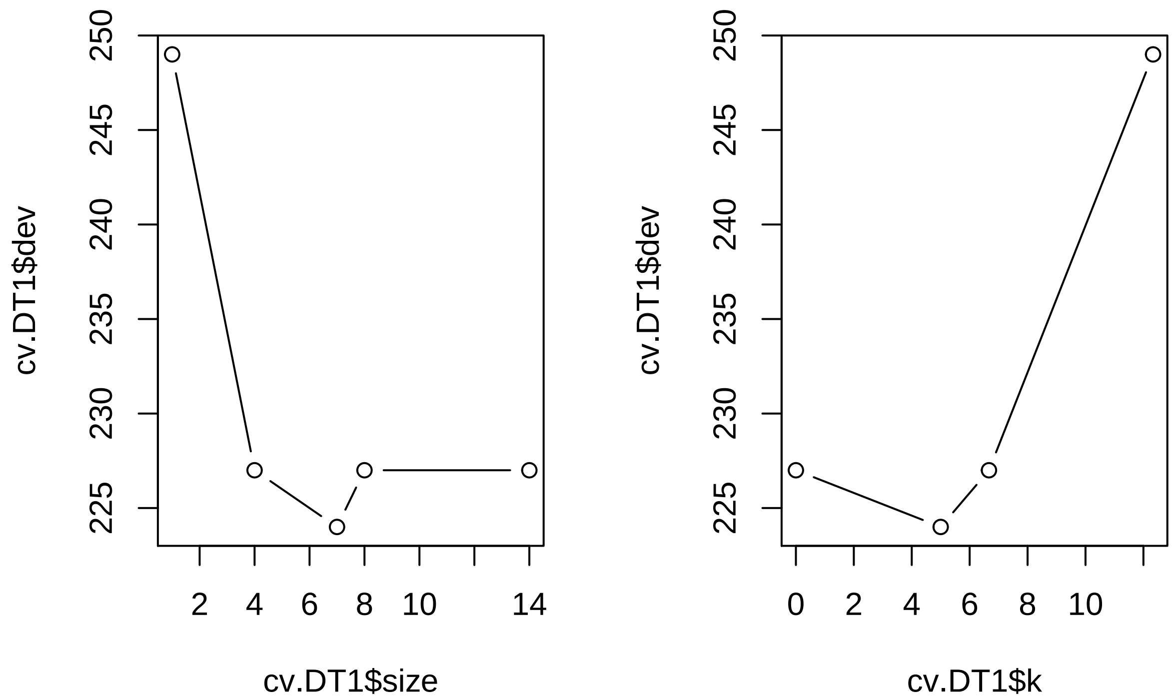
```r
#perform cost complexity pruning by cross-validation (CV) using misclassification rate
set.seed(2021)
cv.DT1 = cv.tree(DT1, FUN=prune.misclass)
```

```r
names(cv.DT1)
```

```
## [1] "size"   "dev"    "k"      "method"
```

Plot the estimated test error rate

```r
par(mfrow = c(1,2))
plot(cv.DT1$size, cv.DT1$dev, type = 'b')
plot(cv.DT1$k, cv.DT1$dev, type = 'b')
```



Get the best size

```r
cv.DT1$size[which.min(cv.DT1$dev)]
```

```
## [1] 7
```

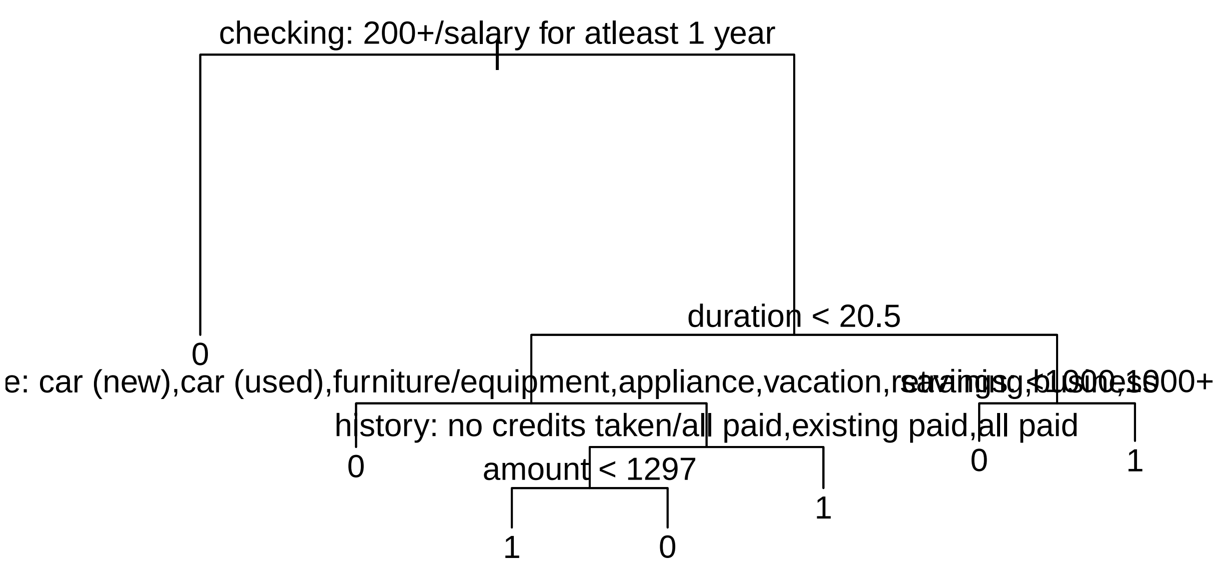
Get the pruned tree of the best size

```r
set.seed(2021)
DT1_pruned = prune.misclass(DT1, best = 7)
summary(DT1_pruned)
```

```
##
## Classification tree:
## snip.tree(tree = DT1, nodes = c(2L, 12L, 15L, 14L))
## Variables actually used in tree construction:
## [1] "checking" "duration" "purpose"  "history"  "amount"   "savings"
## Number of terminal nodes:  7
## Residual mean deviance:  1.006 = 797.7 / 793
## Misclassification error rate: 0.2288 = 183 / 800
```

Plot the pruned tree with 6 leaves

```r
plot(DT1_pruned)
text(DT1_pruned, pretty=0)
```



Get predictions and Confusion Matrix on the test set

```r
test$DT1_pruned.pred = predict(DT1_pruned, test, type = 'class')
caret::confusionMatrix(test$DT1_pruned.pred, test$response)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 109  31
##          1  31  29
##
##                Accuracy : 0.69
##                  95% CI : (0.6209, 0.7533)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 0.6533
##
##                   Kappa : 0.2619
##
##  Mcnemar's Test P-Value : 1.0000
##
##             Sensitivity : 0.7786
##             Specificity : 0.4833
##          Pos Pred Value : 0.7786
##          Neg Pred Value : 0.4833
##              Prevalence : 0.7000
##          Detection Rate : 0.5450
##    Detection Prevalence : 0.7000
##       Balanced Accuracy : 0.6310
##
##        'Positive' Class : 0
##
```

RF1

```r
set.seed(2021)
RF1 <- randomForest(response ~ .,
                                  data = train,
                                  importance = TRUE)
```

```r
#make predictions
test$Pred.RF1 = predict(RF1, test)
caret::confusionMatrix(as.factor(test$Pred.RF1), as.factor(test$response))
```

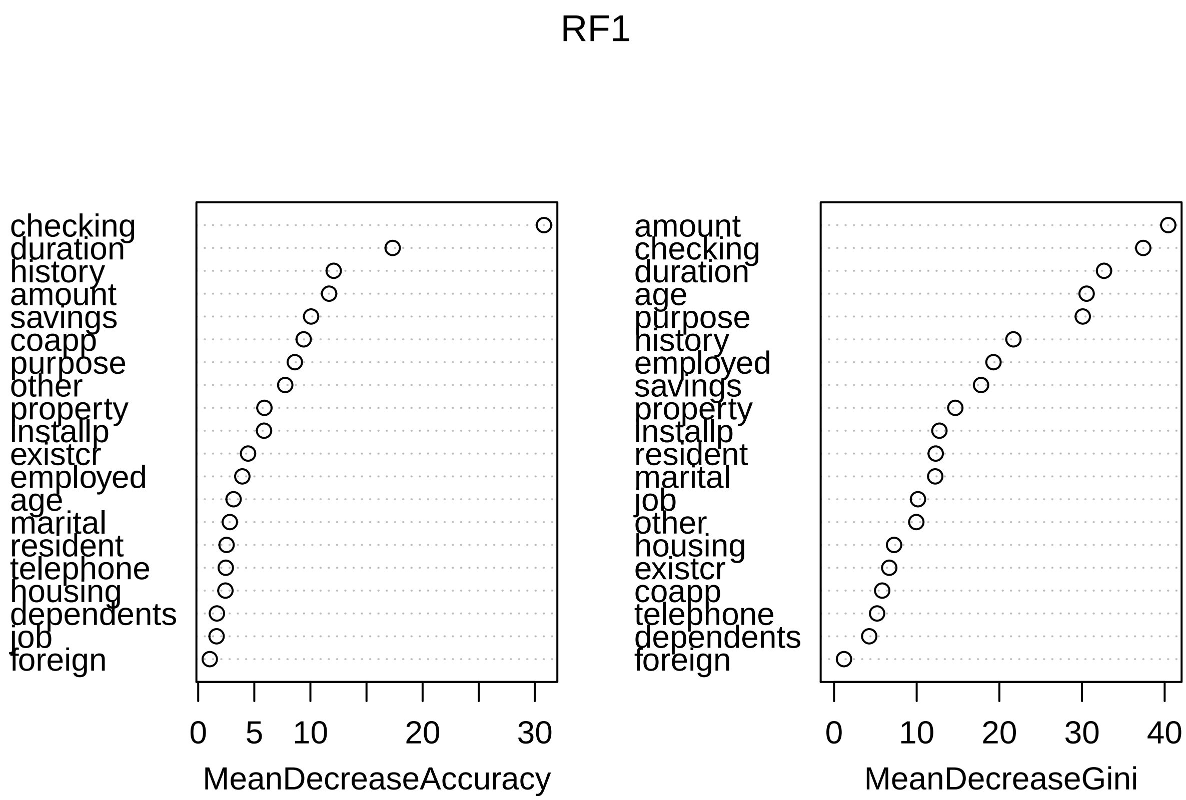
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 132  34
##          1   8  26
##
##                Accuracy : 0.79
```

```
##                  95% CI : (0.7269, 0.8443)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 0.0027247
##
##                   Kappa : 0.4293
##
##  Mcnemar's Test P-Value : 0.0001145
##
##             Sensitivity : 0.9429
##             Specificity : 0.4333
##          Pos Pred Value : 0.7952
##          Neg Pred Value : 0.7647
##              Prevalence : 0.7000
##          Detection Rate : 0.6600
##    Detection Prevalence : 0.8300
##       Balanced Accuracy : 0.6881
##
##        'Positive' Class : 0
##
```

```
#get the variable importance measure for each predictor
importance(RF1)
```

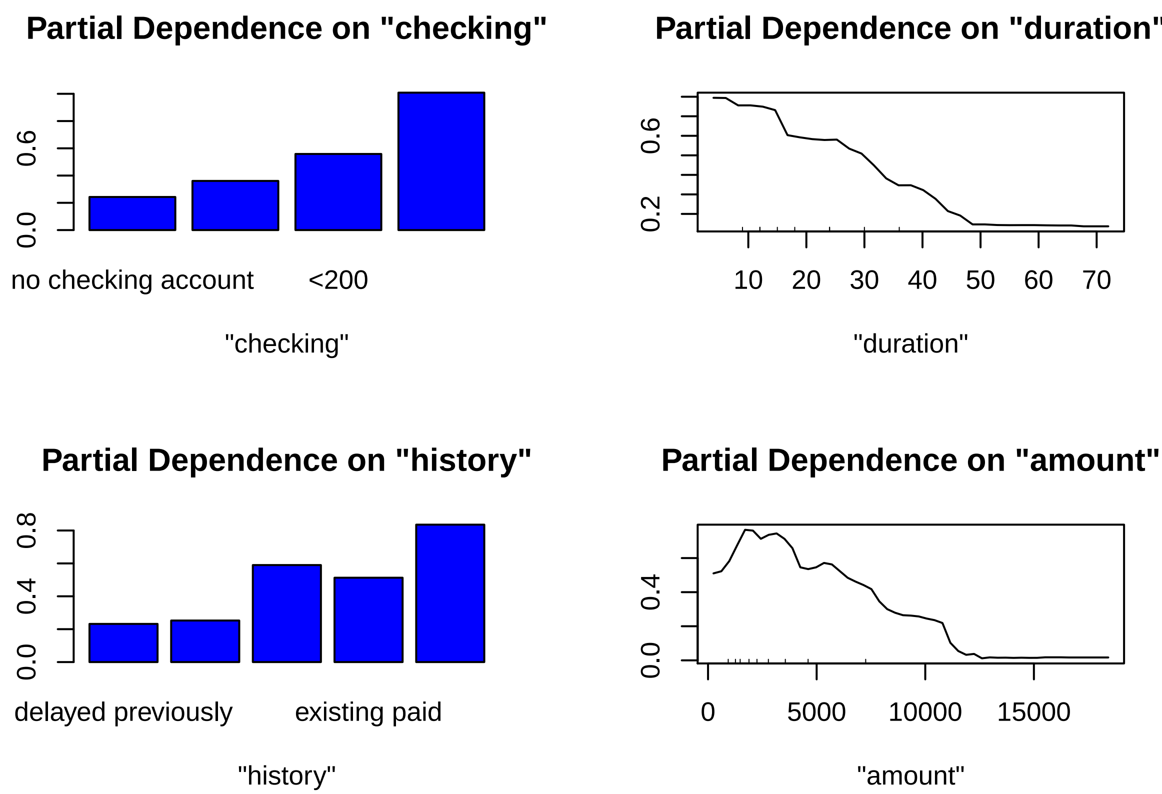
```
##                   0           1 MeanDecreaseAccuracy MeanDecreaseGini
## checking   19.312758 28.26589917            30.815658        37.405200
## duration   14.259420  9.45837449            17.328989        32.666278
## history     8.951702  8.31389335            12.074757        21.706447
## purpose     5.513628  7.40566992             8.611211        30.091545
## amount     10.023344  5.11508087            11.666681        40.432717
## savings     4.584637 10.99113179            10.061678        17.784285
## employed    3.644035  1.65568510             3.931319        19.290627
## installp    5.743730  2.30484864             5.867742        12.748536
## marital     1.374991  2.58016384             2.819629        12.242501
## coapp      10.414430  1.84377793             9.400461         5.824230
## resident    3.013523 -0.06067355             2.525880        12.298480
## property    6.997909 -0.26496353             5.901048        14.673554
## age         2.020129  2.52312503             3.151312        30.558806
## other       7.216453  3.61674689             7.754001         9.953765
## housing     4.116840 -2.01453731             2.424964         7.263702
## existcr     5.443652 -0.61819807             4.443534         6.681547
## job         1.115472  1.18453613             1.637821        10.152922
## dependents  1.612194  0.63198041             1.666016         4.254994
## telephone   1.347248  2.00356421             2.455912         5.208326
## foreign     1.067432  0.31980760             1.031708         1.204413
```

```
varImpPlot(RF1)
```



Partial Dependence Plots

```r
#Method A
par(mfrow=c(2,2))
partialPlot(RF1, pred.data = train, x.var = "checking")
partialPlot(RF1, pred.data = train, x.var = "duration")
partialPlot(RF1, pred.data = train, x.var = "history")
partialPlot(RF1, pred.data = train, x.var = "amount")
```



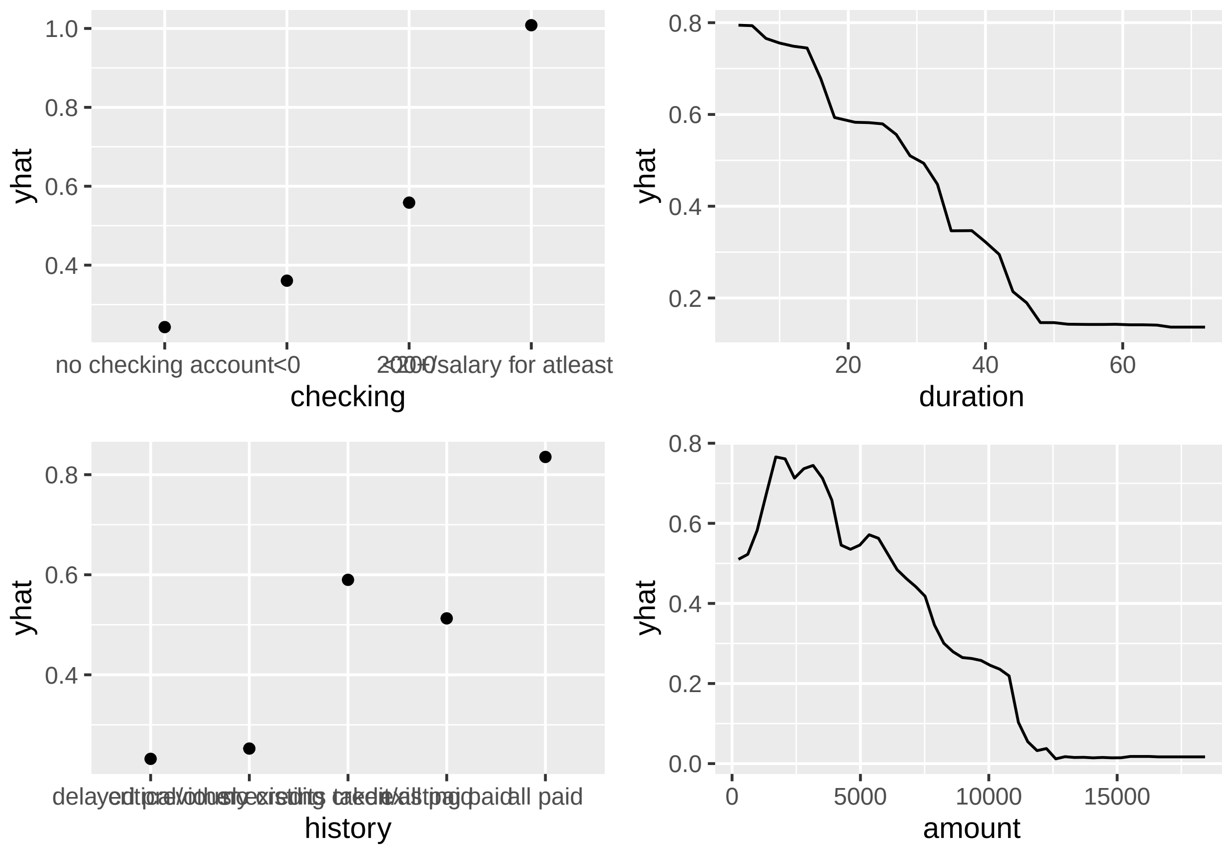

```
#Method B
library(pdp)
library(ggplot2)
par.checking = partial(RF1, pred.var = c("checking"), chull=TRUE)
plot.checking = autoplot(par.checking, contour = T)

par.duration = partial(RF1, pred.var = c("duration"), chull=TRUE)
plot.duration = autoplot(par.duration, contour = T)

par.history = partial(RF1, pred.var = c("history"), chull=TRUE)
plot.history = autoplot(par.history, contour = T)

par.amount = partial(RF1, pred.var = c("amount"), chull=TRUE)
plot.amount = autoplot(par.amount, contour = T)

grid.arrange(plot.checking, plot.duration, plot.history, plot.amount)
```



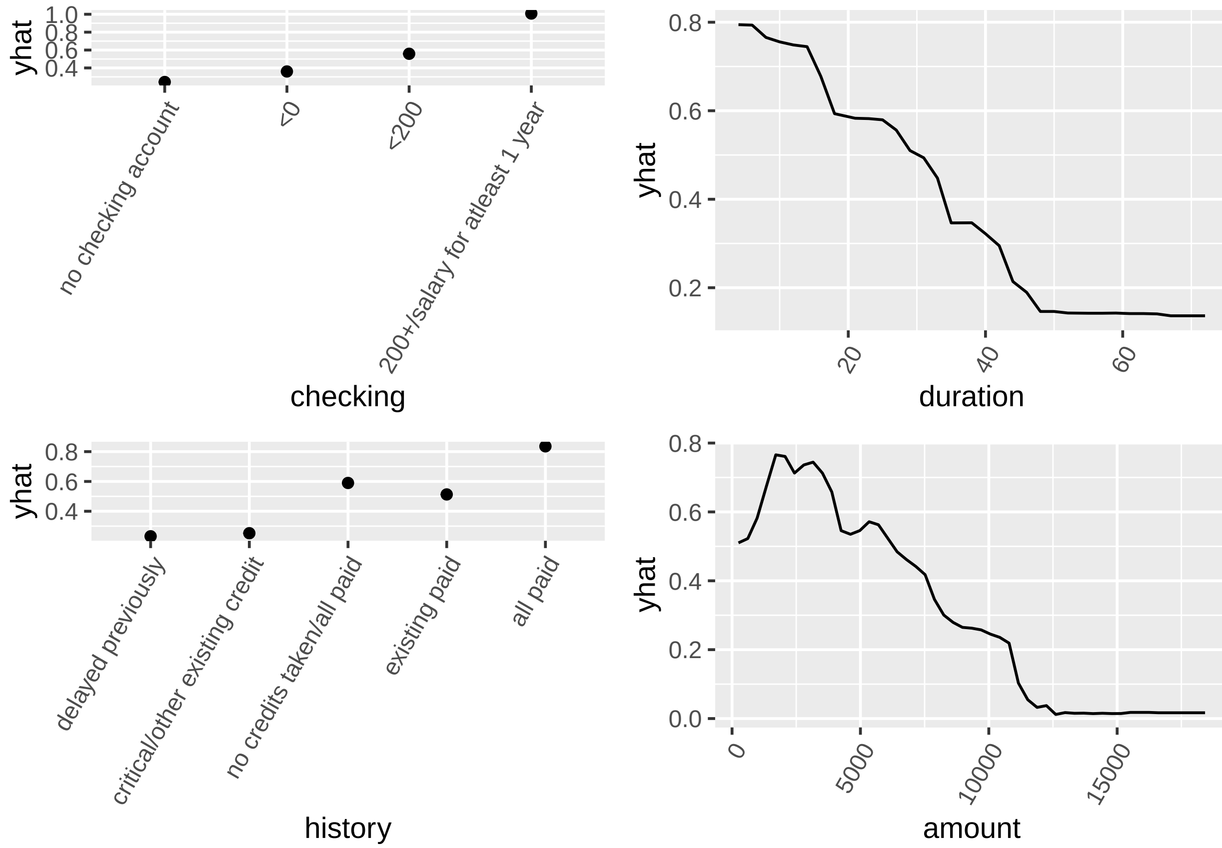

```
#Method B
library(pdp)
library(ggplot2)
par.checking = partial(RF1, pred.var = c("checking"), chull=TRUE)
plot.checking = autoplot(par.checking, contour = T) +
  theme(axis.text.x = element_text(angle = 60, hjust =1))

par.duration = partial(RF1, pred.var = c("duration"), chull=TRUE)
plot.duration = autoplot(par.duration, contour = T) +
  theme(axis.text.x = element_text(angle = 60, hjust =1))

par.history = partial(RF1, pred.var = c("history"), chull=TRUE)
plot.history = autoplot(par.history, contour = T) +
  theme(axis.text.x = element_text(angle = 60, hjust =1))

par.amount = partial(RF1, pred.var = c("amount"), chull=TRUE)
plot.amount = autoplot(par.amount, contour = T) +
  theme(axis.text.x = element_text(angle = 60, hjust =1))

grid.arrange(plot.checking, plot.duration, plot.history, plot.amount)
```



RF2

```r
set.seed(2021)
RF2 <- randomForest(response ~ checking +
                               duration +
                               history +
                               amount +
                               savings +
                               coapp +
                               purpose +
                               other +
                               property +
                               installp,
                               data = train,
                               importance = TRUE)
```

```r
#make predictions
test$Pred.RF2 = predict(RF2, test)
caret::confusionMatrix(as.factor(test$Pred.RF2), as.factor(test$response))
```

```
## Confusion Matrix and Statistics
##
```

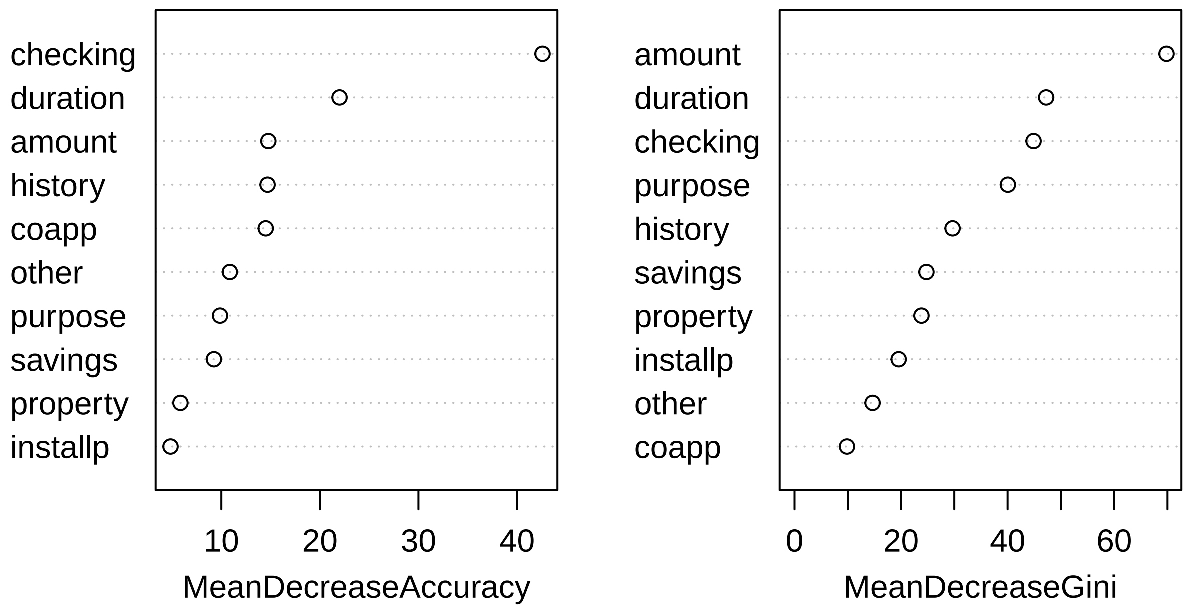
```
##           Reference
## Prediction   0   1
##          0 119  34
##          1  21  26
##
##                Accuracy : 0.725
##                  95% CI : (0.6576, 0.7856)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 0.2455
##
##                   Kappa : 0.302
##
##  Mcnemar's Test P-Value : 0.1056
##
##             Sensitivity : 0.8500
##             Specificity : 0.4333
##          Pos Pred Value : 0.7778
##          Neg Pred Value : 0.5532
##              Prevalence : 0.7000
##          Detection Rate : 0.5950
##    Detection Prevalence : 0.7650
##       Balanced Accuracy : 0.6417
##
##        'Positive' Class : 0
##
```

```r
#get the variable importance measure for each predictor
importance(RF2)
```

```
##                  0          1 MeanDecreaseAccuracy MeanDecreaseGini
## checking 26.198711 34.2338428            42.583180        44.871019
## duration 18.386856  8.5214087            21.993567        47.236745
## history   8.718415 13.2202890            14.688138        29.671937
## amount   14.161127  4.1951056            14.769870        69.822556
## savings   3.904343 11.0799339             9.245003        24.767464
## coapp    15.788349  2.5664961            14.503523         9.850461
## purpose   5.911736  8.7584604             9.864110        40.051124
## other    11.177844  2.9452442            10.861367        14.656931
## property  9.342380 -3.2459821             5.850642        23.828990
## installp  5.342986  0.8821826             4.844372        19.553578
```

```r
varImpPlot(RF2)
```

RF2



RF_TUNED

Hyperparameter Tuning

```
set.seed(2021)
#Create a list of possible values for hyperparameters
mtry.values = seq(2,10,2)
nodesize.values = seq(3,15,3)
ntree.values = seq(2e3, 5e3, 1e3)

#Build a list of possible values for hyperparameters
hyper_grid = expand.grid(mtry = mtry.values, nodesize = nodesize.values, ntree = ntree.values)

#Create an empty vector to store OOB error values
oob_err = c()

#Write a for loop over the rows of hyper_grid to train the grid of models
for (i in 1:nrow(hyper_grid)) {
    model <- randomForest(response ~ ., data = train, importance = T,
                          mtry = hyper_grid$mtry[i],
                          nodesize = hyper_grid$nodesize[i],
                          ntree = hyper_grid$ntree[i])
```

```r
    oob_err[i] <- model$err.rate[length(model$err.rate)] # Store OOB error for the model
}
```

```r
#Identify optimal set of hyperparameters based on OOB error
optimal = which.min(oob_err)
print(hyper_grid[optimal, ])
```

```
##    mtry nodesize ntree
## 35   10        6  3000
```

Tuned hyperparameters: mytr = 10 nodesize = 6 ntree = 3000

Train model with best parameters

```r
set.seed(2021)
RF1_Tuned = randomForest(response ~ .,
                  mtry = 10,
                  nodesize = 6,
                  ntree = 3000,
                  data = train,
                  importance=TRUE)
RF1_Tuned
```

```
##
## Call:
##  randomForest(formula = response ~ ., data = train, mtry = 10,      nodesize = 6, ntree = 3000, impo
##                Type of random forest: classification
##                      Number of trees: 3000
## No. of variables tried at each split: 10
##
##         OOB estimate of  error rate: 23.88%
## Confusion matrix:
##     0   1 class.error
## 0 503  57   0.1017857
## 1 134 106   0.5583333
```

```r
#make predictions
test$Pred.RF1_Tuned = predict(RF1_Tuned, test)
caret::confusionMatrix(as.factor(test$Pred.RF1_Tuned), test$response)
```

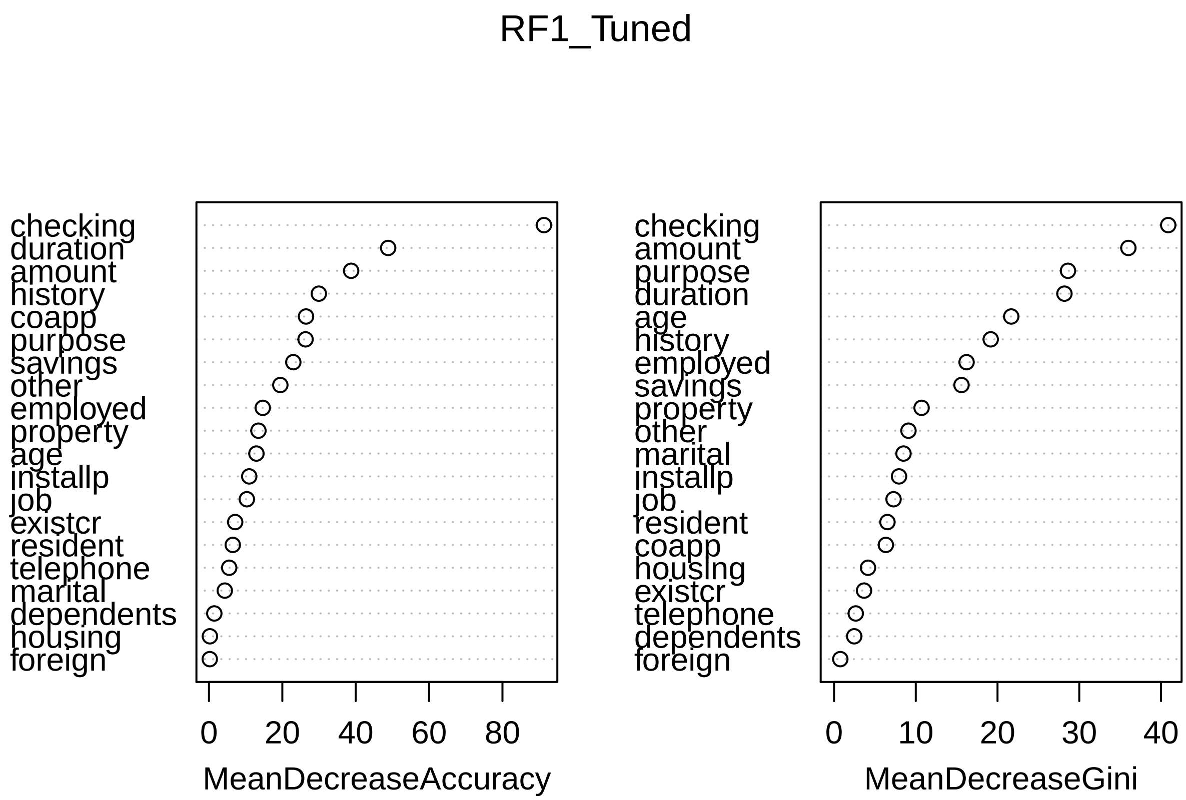
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 127  32
##          1  13  28
##
##                Accuracy : 0.775
##                  95% CI : (0.7108, 0.8309)
##     No Information Rate : 0.7
##     P-Value [Acc > NIR] : 0.01113
##
```

```
##                  Kappa : 0.411
##
##  Mcnemar's Test P-Value : 0.00729
##
##            Sensitivity : 0.9071
##            Specificity : 0.4667
##         Pos Pred Value : 0.7987
##         Neg Pred Value : 0.6829
##             Prevalence : 0.7000
##         Detection Rate : 0.6350
##   Detection Prevalence : 0.7950
##      Balanced Accuracy : 0.6869
##
##       'Positive' Class : 0
##
```

```r
#get the variable importance measure for each predictor
importance(RF1_Tuned)
```

```
##                    0          1 MeanDecreaseAccuracy MeanDecreaseGini
## checking   55.268462 86.7721763           91.3185243       40.8835967
## duration   40.445458 24.0570478           48.8468937       28.1852337
## history    20.067153 22.7565401           29.9513136       19.1668340
## purpose    14.579984 23.8878348           26.3281697       28.6198045
## amount     34.314502 14.4651790           38.7580338       36.0081197
## savings     7.176312 27.5659328           22.9865756       15.5923343
## employed   12.417671  7.6219667           14.6632503       16.2124593
## installp    9.790138  5.5093863           10.9756836        7.9489455
## marital    -2.488411  9.5292116            4.3123988        8.4991468
## coapp      29.176554  2.4071979           26.4605840        6.3367350
## resident    7.009484  1.1479579            6.4889866        6.5339167
## property   18.999537 -4.1878099           13.4820742       10.7092889
## age         9.609500  8.3188499           12.9368176       21.6767371
## other      16.351310 11.5688402           19.4532351        9.1042615
## housing     4.986269 -6.1071748            0.2515763        4.1568885
## existcr    10.542470 -4.3816895            7.1395523        3.6707353
## job        11.367497  1.1622091           10.3282131        7.2801089
## dependents  1.933799 -0.4869316            1.4517844        2.4667056
## telephone   1.894237  5.9703954            5.5141639        2.6553998
## foreign     1.325954 -1.4128011            0.2204540        0.7662337
```

```r
varImpPlot(RF1_Tuned)
```


RF1_Tuned
checking
duration
amount
history
coapp
purpose
savings
other
employed
property
age
installp
job
existcr
resident
telephone
marital
dependents
housing
foreign
0
20
40
60
80
MeanDecreaseAccuracy
checking
amount
purpose
duration
age
history
employed
savings
property
other
marital
installp
job
resident
coapp
housing
existcr
telephone
dependents
foreign
0
10
20
30
40
MeanDecreaseGini