Adult Census Project

2023-05-27

This data was extracted from https://www.kaggle.com/uciml/adult-census-income . The objective of the project is to develop and find the best model that will predict the response variable, income.

Load Data

```
df <- read.csv("adult.csv")
head(df,5)
```

```
##   age workclass fnlwgt    education education.num marital.status
## 1  90         ?  77053      HS-grad             9        Widowed
## 2  82   Private 132870      HS-grad             9        Widowed
## 3  66         ? 186061 Some-college            10        Widowed
## 4  54   Private 140359      7th-8th             4       Divorced
## 5  41   Private 264663 Some-college            10      Separated
##          occupation  relationship  race    sex capital.gain capital.loss
## 1                 ? Not-in-family White Female            0         4356
## 2   Exec-managerial Not-in-family White Female            0         4356
## 3                 ?     Unmarried Black Female            0         4356
## 4 Machine-op-inspct     Unmarried White Female            0         3900
## 5    Prof-specialty     Own-child White Female            0         3900
##   hours.per.week native.country income
## 1             40  United-States  <=50K
## 2             18  United-States  <=50K
## 3             40  United-States  <=50K
## 4             40  United-States  <=50K
## 5             40  United-States  <=50K
```

Dataset overview

```
dim(df)
```

```
## [1] 32561    15
```

Data type for each variable

```
lapply(df, class)
```

```
## $age
## [1] "integer"
##
## $workclass
## [1] "character"
##
## $fnlwgt
## [1] "integer"
```

```
##
## $education
## [1] "character"
##
## $education.num
## [1] "integer"
##
## $marital.status
## [1] "character"
##
## $occupation
## [1] "character"
##
## $relationship
## [1] "character"
##
## $race
## [1] "character"
##
## $sex
## [1] "character"
##
## $capital.gain
## [1] "integer"
##
## $capital.loss
## [1] "integer"
##
## $hours.per.week
## [1] "integer"
##
## $native.country
## [1] "character"
##
## $income
## [1] "character"
```

```r
summary(df)
```

```
##       age         workclass             fnlwgt         education
##  Min.   :17.00   Length:32561       Min.   :  12285   Length:32561
##  1st Qu.:28.00   Class :character   1st Qu.: 117827   Class :character
##  Median :37.00   Mode  :character   Median : 178356   Mode  :character
##  Mean   :38.58                      Mean   : 189778
##  3rd Qu.:48.00                      3rd Qu.: 237051
##  Max.   :90.00                      Max.   :1484705
##  education.num   marital.status      occupation        relationship
##  Min.   : 1.00   Length:32561       Length:32561       Length:32561
##  1st Qu.: 9.00   Class :character   Class :character   Class :character
##  Median :10.00   Mode  :character   Mode  :character   Mode  :character
##  Mean   :10.08
##  3rd Qu.:12.00
##  Max.   :16.00
##      race               sex             capital.gain    capital.loss
##  Length:32561       Length:32561       Min.   :    0   Min.   :   0.0
```

```
##  Class :character   Class :character   1st Qu.:    0   1st Qu.:   0.0
##  Mode  :character   Mode  :character   Median :    0   Median :   0.0
##                                        Mean   : 1078   Mean   :  87.3
##                                        3rd Qu.:    0   3rd Qu.:   0.0
##                                        Max.   :99999   Max.   :4356.0
##  hours.per.week  native.country        income
##  Min.   : 1.00   Length:32561       Length:32561
##  1st Qu.:40.00   Class :character   Class :character
##  Median :40.00   Mode  :character   Mode  :character
##  Mean   :40.44
##  3rd Qu.:45.00
##  Max.   :99.00
```

Handling missing values

```r
df[df == "?"] <- NA
colSums(is.na(df))
```

```
##            age      workclass         fnlwgt      education  education.num
##              0           1836              0              0              0
## marital.status     occupation   relationship           race            sex
##              0           1843              0              0              0
##   capital.gain   capital.loss hours.per.week native.country         income
##              0              0              0            583              0
```

```r
head(df, 5)
```

```
##   age workclass fnlwgt    education education.num marital.status
## 1  90      <NA>  77053      HS-grad             9        Widowed
## 2  82   Private 132870      HS-grad             9        Widowed
## 3  66      <NA> 186061 Some-college            10        Widowed
## 4  54   Private 140359      7th-8th             4       Divorced
## 5  41   Private 264663 Some-college            10      Separated
##          occupation  relationship  race    sex capital.gain capital.loss
## 1              <NA> Not-in-family White Female            0         4356
## 2   Exec-managerial Not-in-family White Female            0         4356
## 3              <NA>     Unmarried Black Female            0         4356
## 4 Machine-op-inspct     Unmarried White Female            0         3900
## 5    Prof-specialty     Own-child White Female            0         3900
##   hours.per.week native.country income
## 1             40  United-States  <=50K
## 2             18  United-States  <=50K
## 3             40  United-States  <=50K
## 4             40  United-States  <=50K
## 5             40  United-States  <=50K
```

```r
mode <- function(x) {                      # Create mode function
  unique_x <- unique(x)
  tabulate_x <- tabulate(match(x, unique_x))
  unique_x[tabulate_x == max(tabulate_x)]
}
mode(df$occupation)
```

```
## [1] "Prof-specialty"
```

Replacing missing values with mode.

```r
df$workclass[is.na(df$workclass)] <- mode(df$workclass)
df$occupation[is.na(df$occupation)] <- mode(df$occupation)
df$native.country[is.na(df$native.country)] <- mode(df$native.country)

summary(df$workclass)
```

```
##    Length     Class      Mode 
##     32561 character character
```

```r
summary(df$occupation)
```

```
##    Length     Class      Mode 
##     32561 character character
```

```r
summary(df$native.country)
```

```
##    Length     Class      Mode 
##     32561 character character
```

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

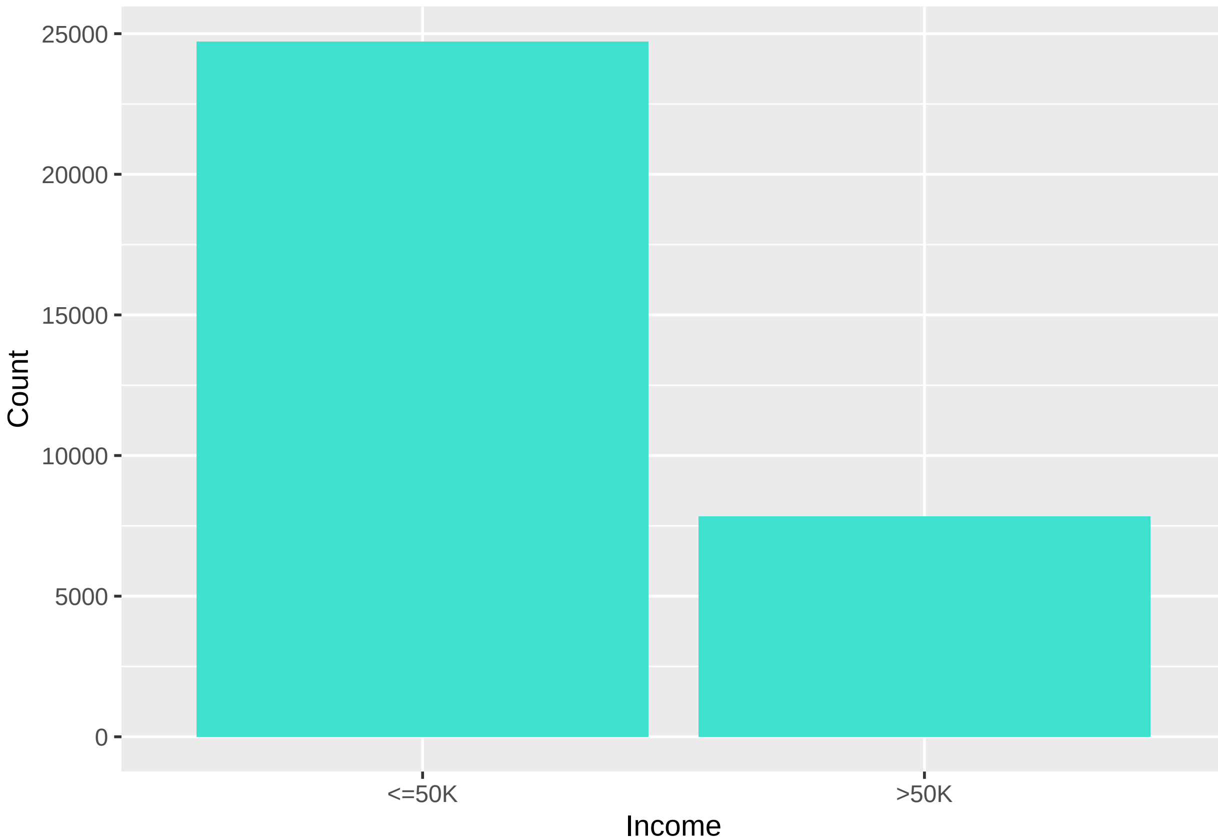
```
##            age      workclass         fnlwgt      education  education.num 
##              0              0              0              0              0 
## marital.status     occupation   relationship           race            sex 
##              0              0              0              0              0 
##   capital.gain   capital.loss hours.per.week native.country         income 
##              0              0              0              0              0
```

```r
head(df,5)
```

```
##   age workclass fnlwgt    education education.num marital.status
## 1  90   Private  77053      HS-grad             9        Widowed
## 2  82   Private 132870      HS-grad             9        Widowed
## 3  66   Private 186061 Some-college            10        Widowed
## 4  54   Private 140359      7th-8th             4       Divorced
## 5  41   Private 264663 Some-college            10      Separated
##          occupation  relationship  race    sex capital.gain capital.loss
## 1    Prof-specialty Not-in-family White Female            0         4356
## 2   Exec-managerial Not-in-family White Female            0         4356
## 3    Prof-specialty     Unmarried Black Female            0         4356
## 4 Machine-op-inspct     Unmarried White Female            0         3900
## 5    Prof-specialty     Own-child White Female            0         3900
##   hours.per.week native.country income
## 1             40  United-States  <=50K
## 2             18  United-States  <=50K
## 3             40  United-States  <=50K
## 4             40  United-States  <=50K
## 5             40  United-States  <=50K
```

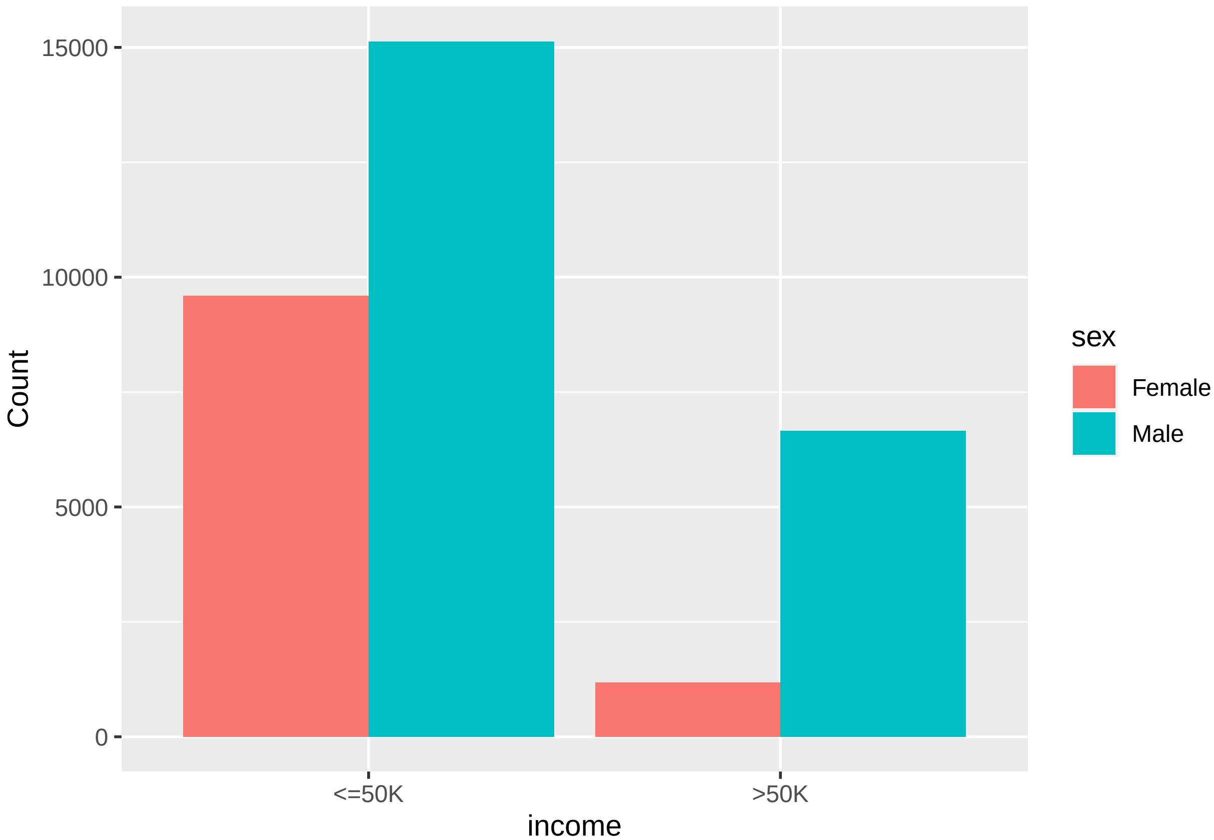
Adult income distribution

```r
library(ggplot2)
ggplot(df) +
  geom_bar(aes(x = income), fill = "turquoise") +
  xlab("Income") + ylab("Count")
```



Adults earning less than 50K are more than two-thirds of the total adults in the census dataset.

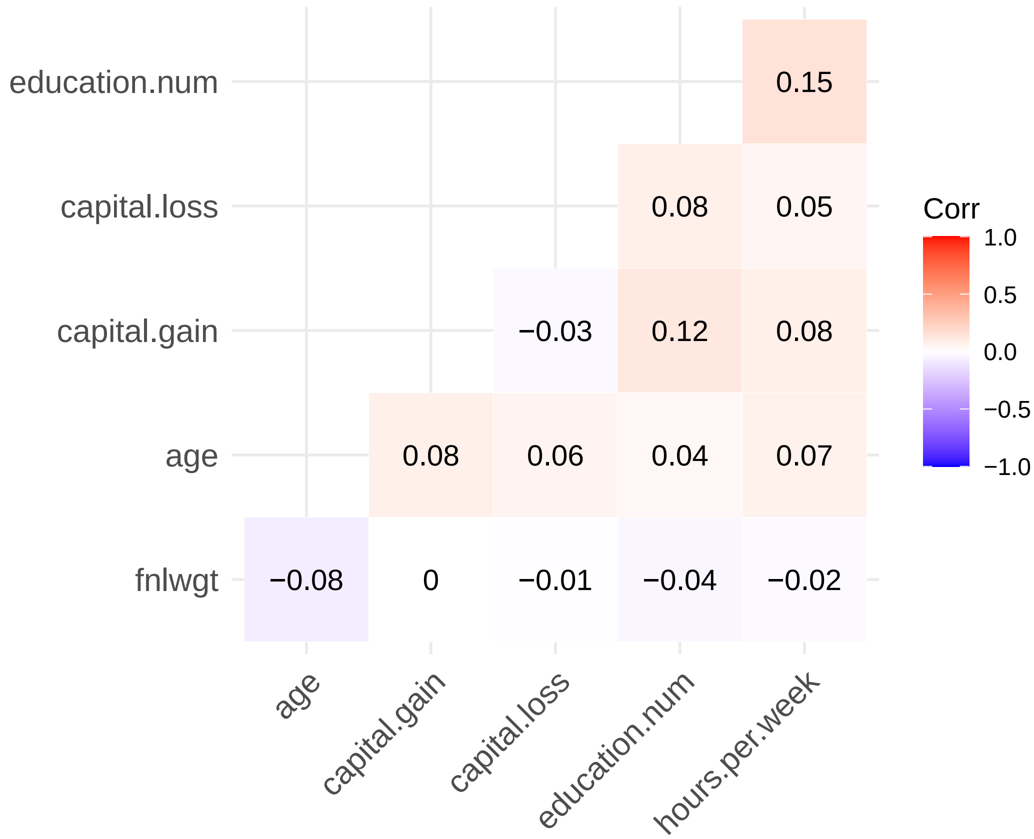
```
ggplot(df) +
  geom_bar(aes(x = income, fill = sex), position = "dodge") +
  xlab("income") + ylab("Count")
```



Males make-up a higher percentage, for both categories of income.

Correlation plot

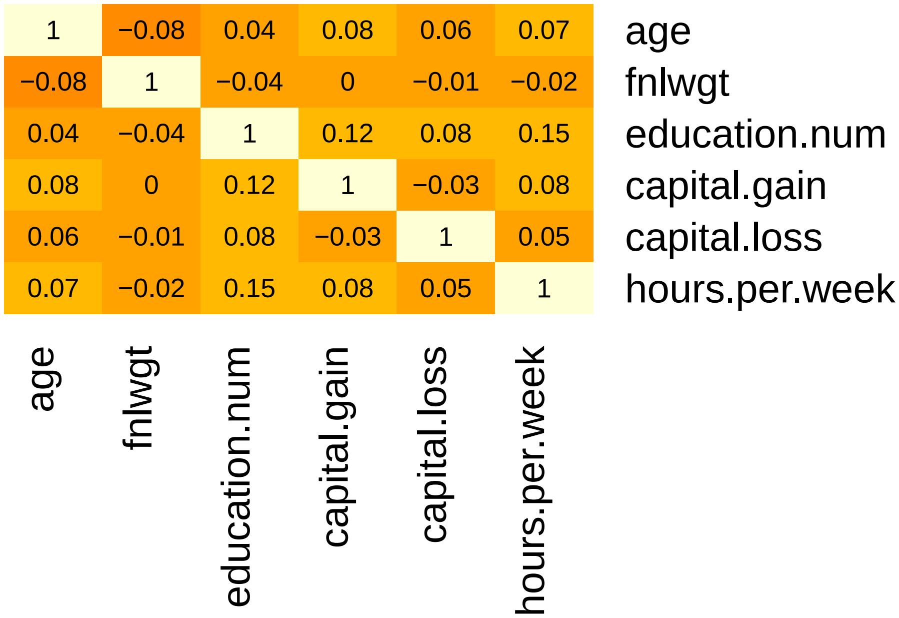
```
library(ggcorrplot)
ggcorrplot(
  cor(df[c("age", "fnlwgt", "education.num", "capital.gain", "capital.loss", "hours.per.week")]),   hc.o
  outline.color = "white",
  lab = TRUE
)
```




```
library(gplots)
```

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

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

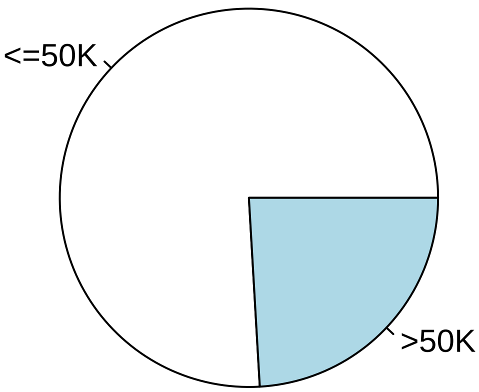
```
df.quant <- df[, c(1, 3, 5, 11, 12, 13)]

heatmap.2(cor(df.quant),
          Rowv = FALSE,
          Colv = FALSE,
          dendrogram = "none",
          cellnote = round(cor(df.quant),2),
          notecol = "black",
          key = FALSE,
          trace = 'none',
          margins = c(10,10))
```



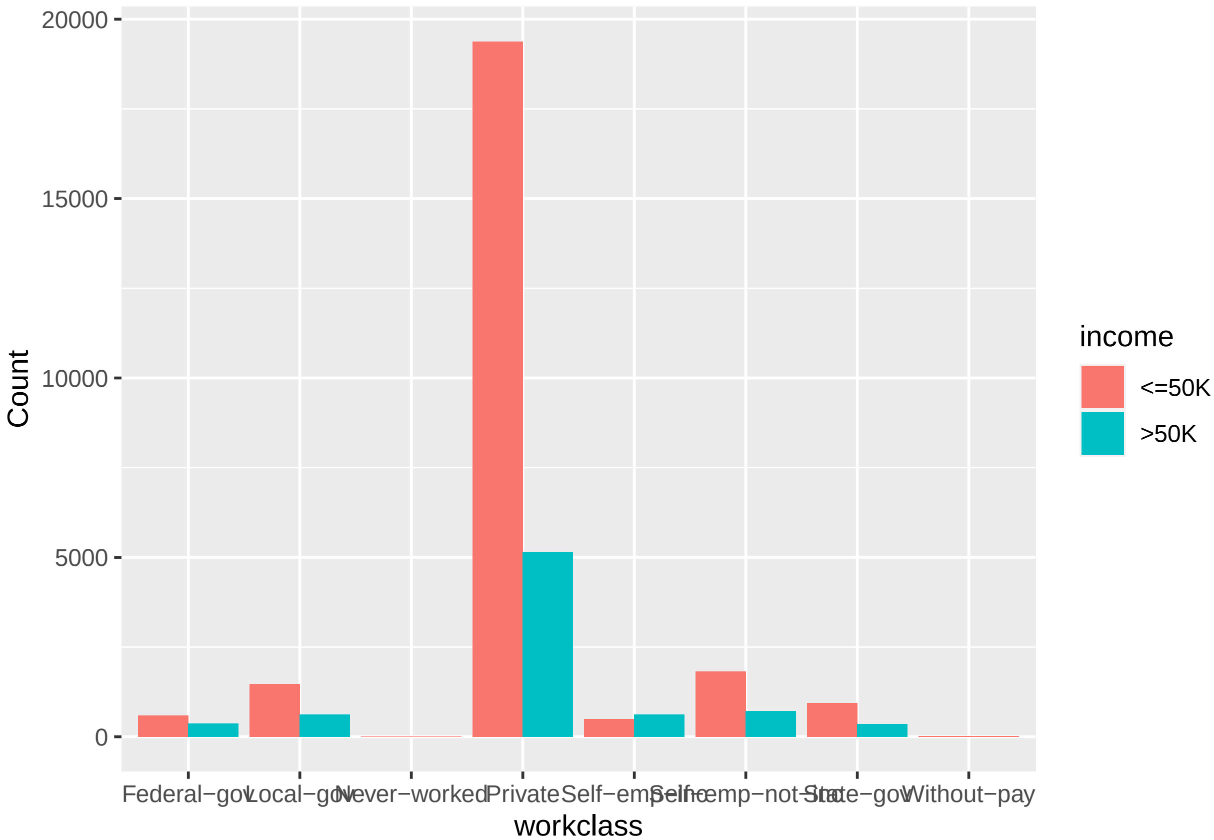

```r
table(df$sex, df$income)
```

```
##
##          <=50K  >50K
##   Female  9592  1179
##   Male   15128  6662
```

```r
pie(table(df$income))
```




```r
ggplot(df) +
  geom_bar(aes(x = workclass, fill = income), position = "dodge") +
  xlab("workclass") + ylab("Count")
```




```r
df$income<-ifelse(df$income==">50K",1,0)
head(df, 5)
```

```
##   age workclass fnlwgt    education education.num marital.status
## 1  90   Private  77053      HS-grad             9        Widowed
## 2  82   Private 132870      HS-grad             9        Widowed
## 3  66   Private 186061 Some-college            10        Widowed
## 4  54   Private 140359      7th-8th             4       Divorced
## 5  41   Private 264663 Some-college            10      Separated
##          occupation  relationship  race    sex capital.gain capital.loss
## 1    Prof-specialty Not-in-family White Female            0         4356
## 2   Exec-managerial Not-in-family White Female            0         4356
## 3    Prof-specialty     Unmarried Black Female            0         4356
## 4 Machine-op-inspct     Unmarried White Female            0         3900
## 5    Prof-specialty     Own-child White Female            0         3900
##   hours.per.week native.country income
## 1             40  United-States      0
## 2             18  United-States      0
## 3             40  United-States      0
## 4             40  United-States      0
## 5             40  United-States      0
```

```r
options(scipen=999)
pcs.cor <- prcomp(df.quant)
summary(pcs.cor)
```

```
## Importance of components:
##                                PC1        PC2        PC3    PC4    PC5    PC6
```

```
## Standard deviation     105549.9778 7385.30256 402.73854 13.64 12.18 2.517
## Proportion of Variance      0.9951    0.00487   0.00001  0.00  0.00 0.000
## Cumulative Proportion       0.9951    0.99999   1.00000  1.00  1.00 1.000
```

```
pcs.cor$rot[,1:4]
```

```
##                          PC1           PC2           PC3            PC4
## age            -0.000009905077 -0.00014351407 0.00201710563  0.96288911067
## fnlwgt          0.999999998721  0.00003043379 0.00003911062  0.00001005091
## education.num  -0.000001052838 -0.00004272285 0.00053293366  0.00987277274
## capital.gain    0.000030367881 -0.99999848346 0.00172935042 -0.00017814585
## capital.loss   -0.000039138988  0.00172989546 0.99999481988 -0.00241554925
## hours.per.week -0.000002195554 -0.00013109584 0.00173646184  0.26970580674
```

PC1 has the highest Proportion of Variance. PC1 is dominated by the variable final weight(fnlwgt) as noted by it having a high scale than other variables.

Adult income prediction models

Logistic regression

```
library(caret)
```

```
## Loading required package: lattice
```

```
library(e1071)
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
set.seed(1)
n = nrow(df)
train.ind <- sample(1:n, n*0.7)

train_data <- df[train.ind,]
valid_data <- df[-train.ind,]

reg <- glm(income ~ ., data = train_data,
           family = "binomial")
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(reg)
```

```
##
## Call:
## glm(formula = income ~ ., family = "binomial", data = train_data)
##
## Coefficients: (1 not defined because of singularities)
##                                          Estimate     Std. Error
## (Intercept)                          -6.8933284473   0.9501919302
## age                                   0.0240744633   0.0019471708
## workclassLocal-gov                   -0.5398694655   0.1340677661
## workclassNever-worked                -9.8544923789 990.0467577329
## workclassPrivate                     -0.5606726752   0.1115868368
## workclassSelf-emp-inc                -0.3360188408   0.1464045619
```

```
## workclassSelf-emp-not-inc                -1.0412410077   0.1315362107
## workclassState-gov                       -0.7870258238   0.1488282925
## workclassWithout-pay                    -14.4591273081 393.5692431918
## fnlwgt                                    0.0000006813   0.0000002072
## education11th                             0.0786381836   0.2465906305
## education12th                             0.2485423799   0.3318889904
## education1st-4th                         -0.1475175242   0.5214975118
## education5th-6th                         -0.3603095094   0.3814619939
## education7th-8th                         -0.7086556198   0.2904704515
## education9th                             -0.2774040493   0.3059064928
## educationAssoc-acdm                       1.2821762542   0.2065538625
## educationAssoc-voc                        1.3136856127   0.1987793398
## educationBachelors                        1.9212542928   0.1839961293
## educationDoctorate                        3.1786846124   0.2556473066
## educationHS-grad                          0.7393737575   0.1792161512
## educationMasters                          2.3488473515   0.1960489896
## educationPreschool                      -21.1407454686 288.4178455327
## educationProf-school                      3.0389377354   0.2362328446
## educationSome-college                     1.1048685685   0.1818726958
## education.num                                       NA             NA
## marital.statusMarried-AF-spouse           2.5580074558   0.6521824794
## marital.statusMarried-civ-spouse          2.1605065238   0.3338707352
## marital.statusMarried-spouse-absent      -0.3429101326   0.3064425063
## marital.statusNever-married              -0.3602458666   0.1062580215
## marital.statusSeparated                   0.0270557273   0.1896036811
## marital.statusWidowed                     0.1647271568   0.1860451287
## occupationArmed-Forces                   -0.8944577690   1.7083351125
## occupationCraft-repair                    0.1515459594   0.0969340563
## occupationExec-managerial                 0.8397429110   0.0935268135
## occupationFarming-fishing                -0.9825830038   0.1685827130
## occupationHandlers-cleaners              -0.6252136872   0.1749348297
## occupationMachine-op-inspct              -0.1820924947   0.1241589781
## occupationOther-service                  -0.8449596269   0.1478683731
## occupationPriv-house-serv                -4.2640450991   1.9692209697
## occupationProf-specialty                  0.2629727088   0.0933923285
## occupationProtective-serv                 0.5367285514   0.1496489092
## occupationSales                           0.3455209525   0.0995044645
## occupationTech-support                    0.7796436653   0.1332610786
## occupationTransport-moving               -0.0201309668   0.1199253926
## relationshipNot-in-family                 0.4412933467   0.3297244768
## relationshipOther-relative               -0.6064148984   0.2993356797
## relationshipOwn-child                    -0.8061604414   0.3283616227
## relationshipUnmarried                     0.3838510551   0.3480020270
## relationshipWife                          1.2510771909   0.1234001613
## raceAsian-Pac-Islander                    0.6968521753   0.3210153501
## raceBlack                                 0.4853680872   0.2819572997
## raceOther                                 0.5914890198   0.4053518006
## raceWhite                                 0.6572144199   0.2689803723
## sexMale                                   0.7965201106   0.0953782304
## capital.gain                              0.0003437860   0.0000128197
## capital.loss                              0.0006391804   0.0000437395
## hours.per.week                            0.0337471261   0.0019640587
## native.countryCanada                     -0.6480750261   0.8683950655
## native.countryChina                      -1.3271345638   0.8854467502
```

```
## native.countryColumbia                     -2.6884020255    1.1807385353
## native.countryCuba                         -0.7074509551    0.8952750801
## native.countryDominican-Republic          -13.7985142458  185.9880237666
## native.countryEcuador                      -1.5819423947    1.3458410665
## native.countryEl-Salvador                  -1.7423736562    1.0188039089
## native.countryEngland                      -0.6213102339    0.8784456059
## native.countryFrance                       -0.4490243477    0.9987299710
## native.countryGermany                      -0.3429673026    0.8636693421
## native.countryGreece                       -1.5178009307    1.0428374623
## native.countryGuatemala                    -1.3029809718    1.3448995603
## native.countryHaiti                         0.1476980090    1.0546139430
## native.countryHoland-Netherlands          -12.2740523388 1455.3977969629
## native.countryHonduras                    -12.7261195247  477.8739964578
## native.countryHong                         -0.1257449974    1.0844396421
## native.countryHungary                      -0.7623965412    1.4938615845
## native.countryIndia                        -0.9041960798    0.8486712383
## native.countryIran                         -0.6773448713    0.9333252992
## native.countryIreland                      -0.0223939866    1.1075047619
## native.countryItaly                         0.2900275407    0.8877157425
## native.countryJamaica                      -1.0306983449    0.9800430470
## native.countryJapan                        -0.9987468646    0.9265047968
## native.countryLaos                         -1.2006933227    1.3399415606
## native.countryMexico                       -1.5650540445    0.8473842335
## native.countryNicaragua                    -2.2123614091    1.3723865032
## native.countryOutlying-US(Guam-USVI-etc)  -13.7100288465  386.3978192026
## native.countryPeru                         -1.7032473500    1.3473306212
## native.countryPhilippines                  -0.2989168647    0.8329722815
## native.countryPoland                       -0.4959072607    0.9417182098
## native.countryPortugal                     -1.4805844740    1.3264830298
## native.countryPuerto-Rico                  -1.2883132338    0.9374024462
## native.countryScotland                     -1.7528207422    1.5643585571
## native.countrySouth                        -1.8089237597    0.9257620723
## native.countryTaiwan                       -0.6793674078    0.9243761810
## native.countryThailand                     -1.6359401080    1.2410515707
## native.countryTrinadad&Tobago              -1.0611963958    1.1963515871
## native.countryUnited-States                -0.6799806629    0.8084058338
## native.countryVietnam                      -1.5324956414    1.0175957221
## native.countryYugoslavia                   -0.5648902069    1.0759215314
##                                          z value             Pr(>|z|)
## (Intercept)                               -7.255  0.00000000000040264 ***
## age                                       12.364 < 0.0000000000000002 ***
## workclassLocal-gov                        -4.027  0.00005653143252096 ***
## workclassNever-worked                     -0.010             0.992058
## workclassPrivate                          -5.025  0.00000050463707760 ***
## workclassSelf-emp-inc                     -2.295             0.021725 *
## workclassSelf-emp-not-inc                 -7.916  0.00000000000000245 ***
## workclassState-gov                        -5.288  0.00000012356195726 ***
## workclassWithout-pay                      -0.037             0.970694
## fnlwgt                                     3.288             0.001010 **
## education11th                              0.319             0.749801
## education12th                              0.749             0.453934
## education1st-4th                          -0.283             0.777274
## education5th-6th                          -0.945             0.344889
## education7th-8th                          -2.440             0.014700 *
```

```
## education9th                          -0.907            0.364499
## educationAssoc-acdm                    6.207  0.00000000053845462 ***
## educationAssoc-voc                     6.609  0.00000000003875437 ***
## educationBachelors                    10.442 < 0.0000000000000002 ***
## educationDoctorate                    12.434 < 0.0000000000000002 ***
## educationHS-grad                       4.126  0.00003697730164749 ***
## educationMasters                      11.981 < 0.0000000000000002 ***
## educationPreschool                    -0.073            0.941568
## educationProf-school                  12.864 < 0.0000000000000002 ***
## educationSome-college                  6.075  0.00000000124021962 ***
## education.num                             NA                  NA
## marital.statusMarried-AF-spouse        3.922  0.00008773442791175 ***
## marital.statusMarried-civ-spouse       6.471  0.00000000009730049 ***
## marital.statusMarried-spouse-absent   -1.119            0.263139
## marital.statusNever-married           -3.390            0.000698 ***
## marital.statusSeparated                0.143            0.886530
## marital.statusWidowed                  0.885            0.375933
## occupationArmed-Forces                -0.524            0.600568
## occupationCraft-repair                 1.563            0.117960
## occupationExec-managerial              8.979 < 0.0000000000000002 ***
## occupationFarming-fishing             -5.828  0.00000000559304586 ***
## occupationHandlers-cleaners           -3.574            0.000352 ***
## occupationMachine-op-inspct           -1.467            0.142483
## occupationOther-service               -5.714  0.00000001101767462 ***
## occupationPriv-house-serv             -2.165            0.030361 *
## occupationProf-specialty               2.816            0.004866 **
## occupationProtective-serv              3.587            0.000335 ***
## occupationSales                        3.472            0.000516 ***
## occupationTech-support                 5.850  0.00000000490103545 ***
## occupationTransport-moving            -0.168            0.866692
## relationshipNot-in-family              1.338            0.180776
## relationshipOther-relative            -2.026            0.042778 *
## relationshipOwn-child                 -2.455            0.014085 *
## relationshipUnmarried                  1.103            0.270021
## relationshipWife                      10.138 < 0.0000000000000002 ***
## raceAsian-Pac-Islander                 2.171            0.029948 *
## raceBlack                              1.721            0.085174 .
## raceOther                              1.459            0.144510
## raceWhite                              2.443            0.014551 *
## sexMale                                8.351 < 0.0000000000000002 ***
## capital.gain                          26.817 < 0.0000000000000002 ***
## capital.loss                          14.613 < 0.0000000000000002 ***
## hours.per.week                        17.182 < 0.0000000000000002 ***
## native.countryCanada                  -0.746            0.455492
## native.countryChina                   -1.499            0.133918
## native.countryColumbia                -2.277            0.022793 *
## native.countryCuba                    -0.790            0.429408
## native.countryDominican-Republic      -0.074            0.940859
## native.countryEcuador                 -1.175            0.239823
## native.countryEl-Salvador             -1.710            0.087226 .
## native.countryEngland                 -0.707            0.479390
## native.countryFrance                  -0.450            0.653002
## native.countryGermany                 -0.397            0.691290
## native.countryGreece                  -1.455            0.145544
```

```
## native.countryGuatemala                    -0.969           0.332629
## native.countryHaiti                         0.140           0.888621
## native.countryHoland-Netherlands           -0.008           0.993271
## native.countryHonduras                     -0.027           0.978754
## native.countryHong                         -0.116           0.907689
## native.countryHungary                      -0.510           0.609804
## native.countryIndia                        -1.065           0.286683
## native.countryIran                         -0.726           0.468003
## native.countryIreland                      -0.020           0.983868
## native.countryItaly                         0.327           0.743886
## native.countryJamaica                      -1.052           0.292943
## native.countryJapan                        -1.078           0.281046
## native.countryLaos                         -0.896           0.370211
## native.countryMexico                       -1.847           0.064758 .
## native.countryNicaragua                    -1.612           0.106950
## native.countryOutlying-US(Guam-USVI-etc)  -0.035           0.971696
## native.countryPeru                         -1.264           0.206171
## native.countryPhilippines                  -0.359           0.719703
## native.countryPoland                       -0.527           0.598473
## native.countryPortugal                     -1.116           0.264348
## native.countryPuerto-Rico                  -1.374           0.169335
## native.countryScotland                     -1.120           0.262512
## native.countrySouth                        -1.954           0.050703 .
## native.countryTaiwan                       -0.735           0.462372
## native.countryThailand                     -1.318           0.187441
## native.countryTrinadad&Tobago              -0.887           0.375064
## native.countryUnited-States                -0.841           0.400271
## native.countryVietnam                      -1.506           0.132068
## native.countryYugoslavia                   -0.525           0.599563
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 25162  on 22791  degrees of freedom
## Residual deviance: 14362  on 22695  degrees of freedom
## AIC: 14556
##
## Number of Fisher Scoring iterations: 14
```

```r
pred <- predict(reg, valid_data, type = "response")

confusionMatrix(
  factor(ifelse(pred > 0.5, 1, 0)),
  factor(valid_data$income),
  positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 6914  986
##          1  502 1367
##
##                Accuracy : 0.8477
```

```
##                  95% CI : (0.8404, 0.8548)
##     No Information Rate : 0.7591
##     P-Value [Acc > NIR] : < 0.00000000000000022
##
##                   Kappa : 0.552
##
##  Mcnemar's Test P-Value : < 0.00000000000000022
##
##             Sensitivity : 0.5810
##             Specificity : 0.9323
##          Pos Pred Value : 0.7314
##          Neg Pred Value : 0.8752
##              Prevalence : 0.2409
##          Detection Rate : 0.1399
##    Detection Prevalence : 0.1913
##       Balanced Accuracy : 0.7566
##
##        'Positive' Class : 1
##
```

Classification Tree

```r
library(rpart)
library(rpart.plot)

set.seed(1)
n = nrow(df)
train.index <- sample(1:n, n * 0.7)
train.data <- df[train.index, ]
valid.data <- df[-train.index, ]


rt <- rpart(as.factor(income) ~ ., data = train.data, method = "class")
summary(rt)
```

```
## Call:
## rpart(formula = as.factor(income) ~ ., data = train.data, method = "class")
##   n= 22792
##
##           CP nsplit rel error    xerror        xstd
## 1 0.12472668      0 1.0000000 1.0000000 0.011761832
## 2 0.06432216      2 0.7505466 0.7505466 0.010585165
## 3 0.03698980      3 0.6862245 0.6862245 0.010216652
## 4 0.01000000      4 0.6492347 0.6492347 0.009990355
##
## Variable importance
##   relationship marital.status   capital.gain      education  education.num
##             24             24             10              9              9
##            sex     occupation            age hours.per.week
##              8              6              6              3
##
## Node number 1: 22792 observations,    complexity param=0.1247267
##   predicted class=0  expected loss=0.2407862  P(node) =1
##     class counts: 17304  5488
```

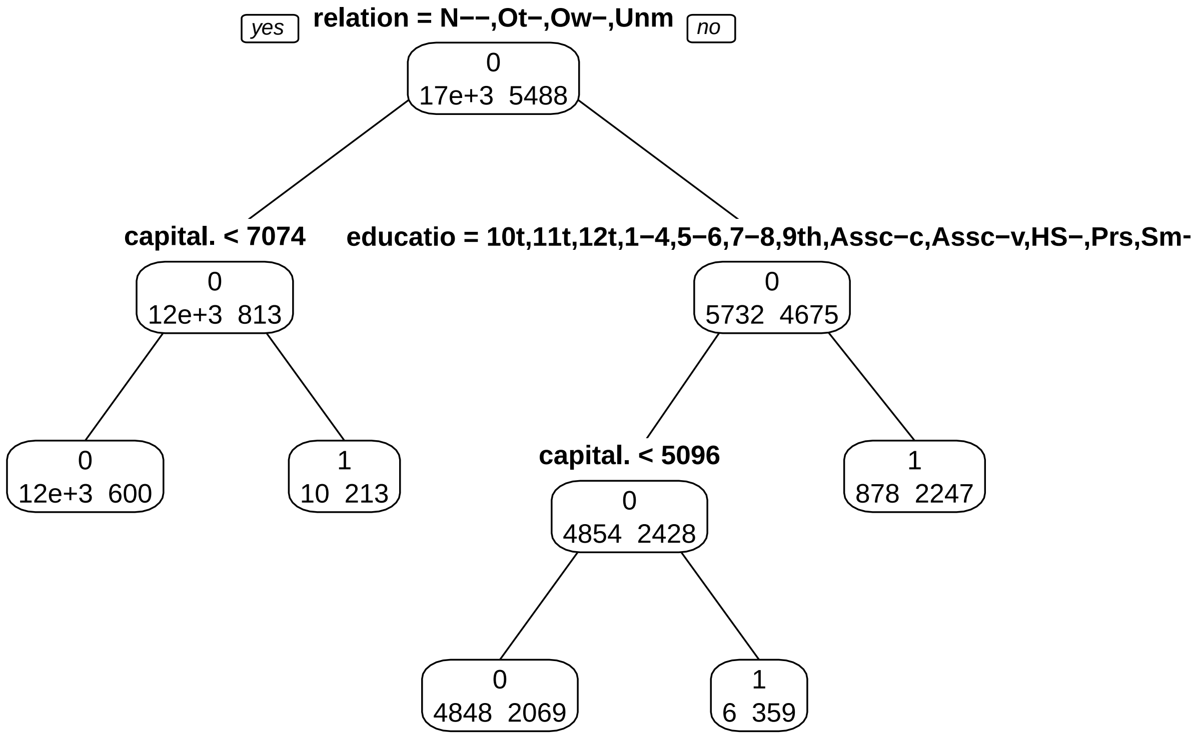
```
##    probabilities: 0.759 0.241
##   left son=2 (12385 obs) right son=3 (10407 obs)
##   Primary splits:
##       relationship   splits as  RLLLLR, improve=1664.0450, (0 missing)
##       marital.status splits as  LRRLLLL, improve=1637.6940, (0 missing)
##       capital.gain   < 5095.5 to the left,  improve=1184.7240, (0 missing)
##       education      splits as  LLLLLLLLLRRLRLRL, improve= 886.0017, (0 missing)
##       education.num  < 12.5   to the left,  improve= 886.0017, (0 missing)
##   Surrogate splits:
##       marital.status splits as  LRRLLLL, agree=0.993, adj=0.984, (0 split)
##       sex            splits as  LR, agree=0.692, adj=0.324, (0 split)
##       age            < 33.5   to the left,  agree=0.649, adj=0.232, (0 split)
##       occupation     splits as  LLRRRLLLLLRLLR, agree=0.621, adj=0.169, (0 split)
##       hours.per.week < 43.5   to the left,  agree=0.603, adj=0.131, (0 split)
##
## Node number 2: 12385 observations,    complexity param=0.0369898
##   predicted class=0  expected loss=0.06564392  P(node) =0.5433924
##     class counts: 11572   813
##    probabilities: 0.934 0.066
##   left son=4 (12162 obs) right son=5 (223 obs)
##   Primary splits:
##       capital.gain   < 7073.5 to the left,  improve=359.36060, (0 missing)
##       education      splits as  LLLLLLLLLLLRLRLRL, improve=104.34020, (0 missing)
##       education.num  < 13.5   to the left,  improve=104.34020, (0 missing)
##       hours.per.week < 42.5   to the left,  improve= 74.63036, (0 missing)
##       occupation     splits as  LLLRLLLLLRRLRL, improve= 52.13196, (0 missing)
##
## Node number 3: 10407 observations,    complexity param=0.1247267
##   predicted class=0  expected loss=0.4492169  P(node) =0.4566076
##     class counts:  5732  4675
##    probabilities: 0.551 0.449
##   left son=6 (7282 obs) right son=7 (3125 obs)
##   Primary splits:
##       education     splits as  LLLLLLLLLLRRLRLRL, improve=650.2992, (0 missing)
##       education.num < 12.5   to the left,  improve=650.2992, (0 missing)
##       capital.gain  < 5095.5 to the left,  improve=543.4804, (0 missing)
##       occupation    splits as  LRLRLLLLLRRRRL, improve=538.9977, (0 missing)
##       capital.loss  < 1782.5 to the left,  improve=181.1397, (0 missing)
##   Surrogate splits:
##       education.num  < 12.5   to the left,  agree=1.000, adj=1.000, (0 split)
##       occupation     splits as  LLLRLLLLLRLLLL, agree=0.770, adj=0.233, (0 split)
##       capital.gain   < 7493   to the left,  agree=0.717, adj=0.058, (0 split)
##       native.country splits as  LLRLLLLLRRLLLL-LRLRRLLLRLLLLLRLLLLLRLLLLL, agree=0.707, adj=0.025, (0
##       capital.loss   < 1894.5 to the left,  agree=0.704, adj=0.015, (0 split)
##
## Node number 4: 12162 observations
##   predicted class=0  expected loss=0.04933399  P(node) =0.5336083
##     class counts: 11562   600
##    probabilities: 0.951 0.049
##
## Node number 5: 223 observations
##   predicted class=1  expected loss=0.04484305  P(node) =0.009784135
##     class counts:    10   213
##    probabilities: 0.045 0.955
```

```
##
## Node number 6: 7282 observations,    complexity param=0.06432216
##   predicted class=0  expected loss=0.3334249  P(node) =0.3194981
##     class counts:  4854  2428
##    probabilities: 0.667 0.333
##   left son=12 (6917 obs) right son=13 (365 obs)
##   Primary splits:
##       capital.gain  < 5095.5 to the left,  improve=324.83680, (0 missing)
##       occupation    splits as  RLLRLLLLLLRRRL, improve=141.42320, (0 missing)
##       education     splits as  LLLLLLLRR--R-L-R, improve=125.67990, (0 missing)
##       education.num < 8.5    to the left,  improve=125.67990, (0 missing)
##       capital.loss  < 1782.5 to the left,  improve= 87.49491, (0 missing)
##
## Node number 7: 3125 observations
##   predicted class=1  expected loss=0.28096  P(node) =0.1371095
##     class counts:   878  2247
##    probabilities: 0.281 0.719
##
## Node number 12: 6917 observations
##   predicted class=0  expected loss=0.2991181  P(node) =0.3034837
##     class counts:  4848  2069
##    probabilities: 0.701 0.299
##
## Node number 13: 365 observations
##   predicted class=1  expected loss=0.01643836  P(node) =0.01601439
##     class counts:     6   359
##    probabilities: 0.016 0.984
```

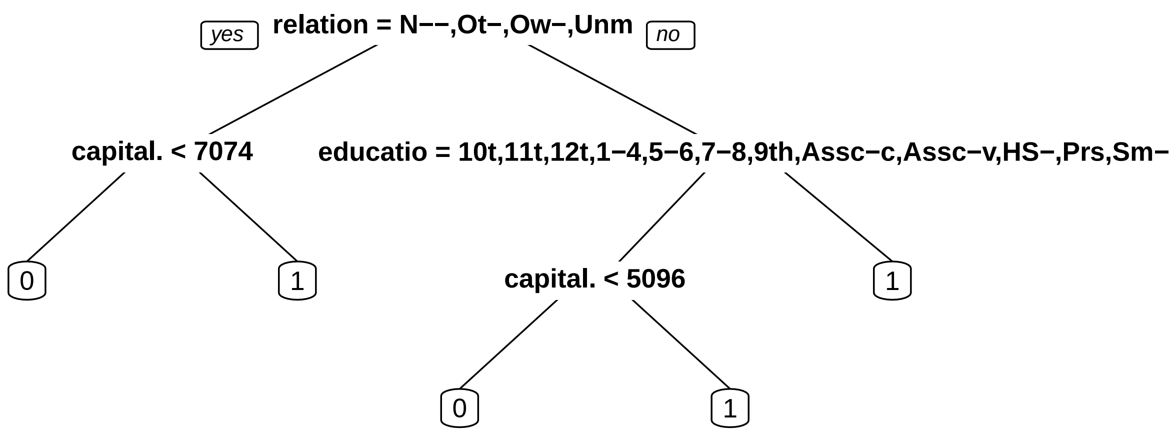
```r
rt$variable.importance
```

```
##   relationship marital.status   capital.gain      education  education.num
##    1664.045007    1637.022464     721.862747     650.299228     650.299228
##            sex     occupation            age hours.per.week native.country
##     539.971172     433.599694     386.150542     218.738692      16.439564
##   capital.loss
##       9.572405
```

```r
prp(rt, type = 1, extra = 1)
```




```
prp(rt)
```




```
pred.train <- predict(rt, valid.data)
```

```
set.seed(1)
n = nrow(df)
train.index <- sample(1:n, n * 0.7)
train.data <- df[train.index, ]
valid.data <- df[-train.index, ]

rt <- rpart(as.factor(income) ~., data = train.data, method = "class")

rt.pred <- predict(rt, valid.data, type = "class")

confusionMatrix(rt.pred,
                as.factor(valid.data$income),
                positive = "1"
```

```
)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 7063 1147
##          1  353 1206
##
##                Accuracy : 0.8465
##                  95% CI : (0.8391, 0.8535)
##     No Information Rate : 0.7591
##     P-Value [Acc > NIR] : < 0.00000000000000022
##
##                   Kappa : 0.5255
##
##  Mcnemar's Test P-Value : < 0.00000000000000022
##
##             Sensitivity : 0.5125
##             Specificity : 0.9524
##          Pos Pred Value : 0.7736
##          Neg Pred Value : 0.8603
##              Prevalence : 0.2409
##          Detection Rate : 0.1235
##    Detection Prevalence : 0.1596
##       Balanced Accuracy : 0.7325
##
##        'Positive' Class : 1
##
```

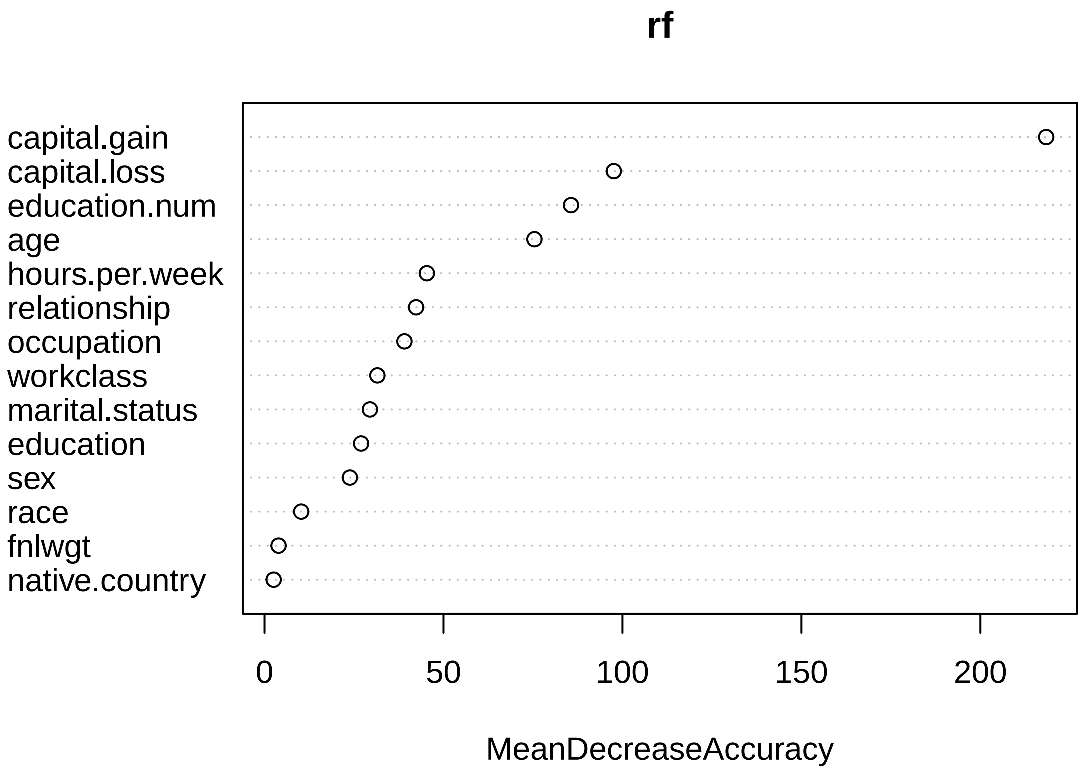
Random Forest

```
library(randomForest)
```

```
## randomForest 4.7-1.1

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
set.seed(1)
n = nrow(df)
train.index <- sample(1:n, n * 0.7)
train.data <- df[train.index, ]
valid.data <- df[-train.index, ]


rf <- randomForest(as.factor(income)~ .,
                   data = train.data, ntree = 500,
                   mtry = 4, nodesize = 5, importance = TRUE
)
```

```r
## variable importance plot
varImpPlot(rf, type = 1)
```




```r
## confusion matrix
rf.pred <- predict(rf, valid.data)
confusionMatrix(rf.pred, as.factor(valid.data$income),
                positive = "1"
)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 7024  956
##          1  392 1397
##
##                Accuracy : 0.862
##                  95% CI : (0.855, 0.8688)
##     No Information Rate : 0.7591
##     P-Value [Acc > NIR] : < 0.00000000000000022
##
##                   Kappa : 0.589
##
##  Mcnemar's Test P-Value : < 0.00000000000000022
##
##             Sensitivity : 0.5937
##             Specificity : 0.9471
##          Pos Pred Value : 0.7809
##          Neg Pred Value : 0.8802
```

```
##             Prevalence : 0.2409
##         Detection Rate : 0.1430
##   Detection Prevalence : 0.1831
##      Balanced Accuracy : 0.7704
##
##       'Positive' Class : 1
##
```

Model evaluation

```r
Models <- data.frame(
         Model = c("Logistic Regression", "Classification Tree",
                   "Random Forest"),
         Accuracy = c(84.77, 84.65, 86.22 )
)
knitr::kable(Models, "pipe", col.name=c("Model", "Accuracy"), align = c("l", "c"))
```

Model	Accuracy
Logistic Regression	84.77
Classification Tree	84.65
Random Forest	86.22

From the table above, Random Forest has the best accuracy.