Income Logistic Regression Project

Kiro Shenouda

4/14/2022

My goal for this project is to create a classifier that predicts whether someone makes an annual income of below or above 50,000 dollars. I will create the classifier using logistic regression.

Get the Data

```
adult = read.csv('adult_sal.csv')
head(adult)
```

```
##   X age    type_employer fnlwgt education education_num            marital
## 1 1  39        State-gov  77516 Bachelors            13      Never-married
## 2 2  50 Self-emp-not-inc  83311 Bachelors            13 Married-civ-spouse
## 3 3  38          Private 215646   HS-grad             9           Divorced
## 4 4  53          Private 234721      11th             7 Married-civ-spouse
## 5 5  28          Private 338409 Bachelors            13 Married-civ-spouse
## 6 6  37          Private 284582   Masters            14 Married-civ-spouse
##          occupation  relationship  race    sex capital_gain capital_loss
## 1      Adm-clerical Not-in-family White   Male         2174            0
## 2   Exec-managerial       Husband White   Male            0            0
## 3 Handlers-cleaners Not-in-family White   Male            0            0
## 4 Handlers-cleaners       Husband Black   Male            0            0
## 5    Prof-specialty          Wife Black Female            0            0
## 6   Exec-managerial          Wife White Female            0            0
##   hr_per_week       country income
## 1          40 United-States  <=50K
## 2          13 United-States  <=50K
## 3          40 United-States  <=50K
## 4          40 United-States  <=50K
## 5          40          Cuba  <=50K
## 6          40 United-States  <=50K
```

```
summary(adult)
```

```
##        X              age        type_employer          fnlwgt
##  Min.   :    1   Min.   :17.00   Length:32561       Min.   :  12285
##  1st Qu.: 8141   1st Qu.:28.00   Class :character   1st Qu.: 117827
##  Median :16281   Median :37.00   Mode  :character   Median : 178356
##  Mean   :16281   Mean   :38.58                      Mean   : 189778
##  3rd Qu.:24421   3rd Qu.:48.00                      3rd Qu.: 237051
```

```
##  Max.   :32561   Max.   :90.00                      Max.   :1484705
##   education         education_num     marital           occupation
##  Length:32561       Min.   : 1.00   Length:32561       Length:32561
##  Class :character   1st Qu.: 9.00   Class :character   Class :character
##  Mode  :character   Median :10.00   Mode  :character   Mode  :character
##                     Mean   :10.08
##                     3rd Qu.:12.00
##                     Max.   :16.00
##  relationship           race               sex             capital_gain
##  Length:32561       Length:32561       Length:32561       Min.   :    0
##  Class :character   Class :character   Class :character   1st Qu.:    0
##  Mode  :character   Mode  :character   Mode  :character   Median :    0
##                                                           Mean   : 1078
##                                                           3rd Qu.:    0
##                                                           Max.   :99999
##   capital_loss     hr_per_week      country             income
##  Min.   :   0.0   Min.   : 1.00   Length:32561       Length:32561
##  1st Qu.:   0.0   1st Qu.:40.00   Class :character   Class :character
##  Median :   0.0   Median :40.00   Mode  :character   Mode  :character
##  Mean   :  87.3   Mean   :40.44
##  3rd Qu.:   0.0   3rd Qu.:45.00
##  Max.   :4356.0   Max.   :99.00
```

```
str(adult)
```

```
## 'data.frame':    32561 obs. of  16 variables:
##  $ X            : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ age          : int  39 50 38 53 28 37 49 52 31 42 ...
##  $ type_employer: chr  "State-gov" "Self-emp-not-inc" "Private" "Private" ...
##  $ fnlwgt       : int  77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 ...
##  $ education    : chr  "Bachelors" "Bachelors" "HS-grad" "11th" ...
##  $ education_num: int  13 13 9 7 13 14 5 9 14 13 ...
##  $ marital      : chr  "Never-married" "Married-civ-spouse" "Divorced" "Married-civ-spouse" ...
##  $ occupation   : chr  "Adm-clerical" "Exec-managerial" "Handlers-cleaners" "Handlers-cleaners" ...
##  $ relationship : chr  "Not-in-family" "Husband" "Not-in-family" "Husband" ...
##  $ race         : chr  "White" "White" "White" "Black" ...
##  $ sex          : chr  "Male" "Male" "Male" "Male" ...
##  $ capital_gain : int  2174 0 0 0 0 0 0 0 14084 5178 ...
##  $ capital_loss : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr_per_week  : int  40 13 40 40 40 40 16 45 50 40 ...
##  $ country      : chr  "United-States" "United-States" "United-States" "United-States" ...
##  $ income       : chr  "<=50K" "<=50K" "<=50K" "<=50K" ...
```

Data Cleaning

```
table(adult$type_employer)
```

```
##
##                ?      Federal-gov        Local-gov     Never-worked
##             1836              960             2093                7
##          Private     Self-emp-inc Self-emp-not-inc        State-gov
##            22696             1116             2541             1298
##      Without-pay
##               14
```

I will clean these columns by combining some of them together, thus reducing the number of factors used for classification.

```
unemployed = function(job) {
  job = as.character(job)
  if(job == 'Never-worked' | job == 'Without-pay') {
    return('Unemployed')
  } else {
    return(job)
  }
}
adult$type_employer = sapply(adult$type_employer, unemployed)

selfemploy = function(job) {
  job = as.character(job)
  if(job == 'Self-emp-inc' | job == 'Self-emp-not-inc') {
    return('Self-emp')
  } else {
    return(job)
  }
}
adult$type_employer = sapply(adult$type_employer, selfemploy)

local = function(job) {
    job = as.character(job)
    if(job == 'Local-gov' | job == 'State-gov') {
        return('SL-gov')
    } else {
        return(job)
    }
}
adult$type_employer = sapply(adult$type_employer, local)

table(adult$type_employer)
```

```
##
##           ? Federal-gov     Private    Self-emp      SL-gov  Unemployed
##        1836         960       22696        3657        3391          21
```

```
table(adult$marital)
```

```
##
##              Divorced     Married-AF-spouse    Married-civ-spouse
##                  4443                    23                 14976
## Married-spouse-absent         Never-married             Separated
##                   418                 10683                  1025
##               Widowed
##                   993
```

Like the employment type column, I will clean the marital column by combining some of the columns together.

```
married = function(status) {
    status = as.character(status)
    if(status == 'Married-AF-spouse' |
       status == 'Married-civ-spouse' |
       status == 'Married-spouse-absent') {
        return('Married')
    } else if(status == 'Divorced' |
              status == 'Separated' |
              status == 'Widowed') {
        return('Not-Married')
    } else {
        return(status)
    }
}
adult$marital = sapply(adult$marital, married)

table(adult$marital)
```

```
##
##       Married Never-married   Not-Married
##         15417         10683          6461
```

```
table(adult$country)
```

```
##
##                          ?                   Cambodia
##                        583                         19
##                     Canada                      China
##                        121                         75
##                   Columbia                       Cuba
##                         59                         95
##         Dominican-Republic                    Ecuador
##                         70                         28
##                El-Salvador                    England
##                        106                         90
##                     France                    Germany
##                         29                        137
##                     Greece                  Guatemala
##                         29                         64
```

```
##                        Haiti         Holand-Netherlands
##                           44                          1
##                     Honduras                       Hong
##                           13                         20
##                      Hungary                      India
##                           13                        100
##                         Iran                    Ireland
##                           43                         24
##                        Italy                    Jamaica
##                           73                         81
##                        Japan                       Laos
##                           62                         18
##                       Mexico                  Nicaragua
##                          643                         34
## Outlying-US(Guam-USVI-etc)                       Peru
##                           14                         31
##                  Philippines                     Poland
##                          198                         60
##                     Portugal                Puerto-Rico
##                           37                        114
##                     Scotland                      South
##                           12                         80
##                       Taiwan                   Thailand
##                           51                         18
##              Trinadad&Tobago              United-States
##                           19                      29170
##                      Vietnam                 Yugoslavia
##                           67                         16
```

I will reduce the number of countries by grouping countries of the same continent/region together.

```
Asia = c('Cambodia', 'China', 'Hong', 'India', 'Iran', 'Japan',
         'Laos', 'Philippines', 'Taiwan', 'Thailand', 'Vietnam')
North.America = c('Canada', 'Puerto-Rico', 'United-States')
Europe = c('England', 'France', 'Germany', 'Greece',
           'Holand-Netherlands', 'Hungary', 'Ireland', 'Italy',
           'Poland', 'Portugal', 'Scotland', 'Yugoslavia')
Latin.and.South.America = c('Columbia', 'Cuba',
                            'Dominican-Republic', 'Ecuador',
                            'El-Salvador', 'Guatemala', 'Haiti',
                            'Honduras', 'Mexico', 'Nicaragua',
                            'Outlying-US(Guam-USVI-etc)', 'Peru',
                            'Jamaica', 'Trinadad&Tobago')
Other = c('South')

region = function(country) {
    if(country %in% Asia) {
        return('Asia')
    } else if(country %in% North.America) {
        return('North.America')
    } else if(country %in% Europe) {
        return('Europe')
    } else if(country %in% Latin.and.South.America) {
        return('Latin.and.South.America')
```

```
    } else {
        return('Other')
    }
}
adult$country = sapply(adult$country, region)

table(adult$country)
```

```
##
##                  Asia                 Europe Latin.and.South.America
##                   671                    521                    1301
##         North.America                  Other
##                 29405                    663
```

Since most of the variables are still continuous, I will convert them to be factors for the classification model.

```
adult$type_employer = factor(adult$type_employer)
adult$education = factor(adult$education)
adult$marital = factor(adult$marital)
adult$occupation = factor(adult$occupation)
adult$relationship = factor(adult$relationship)
adult$race = factor(adult$race)
adult$sex = factor(adult$sex)
adult$country = factor(adult$country)
adult$income = factor(adult$income)
```

Missing Data

```r
adult[adult == '?' | adult == ' ?'] = NA
adult = na.omit(adult)
adult = adult[,-1]
names(adult)[names(adult) == 'country'] = 'region'
str(adult)
```

```
## 'data.frame':    30718 obs. of  15 variables:
##  $ age          : int  39 50 38 53 28 37 49 52 31 42 ...
##  $ type_employer: Factor w/ 6 levels "?","Federal-gov",..: 5 4 3 3 3 3 3 4 3 3 ...
##  $ fnlwgt       : int  77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 ...
##  $ education    : Factor w/ 16 levels "10th","11th",..: 10 10 12 2 10 13 7 12 13 10 ...
##  $ education_num: int  13 13 9 7 13 14 5 9 14 13 ...
##  $ marital      : Factor w/ 3 levels "Married","Never-married",..: 2 1 3 1 1 1 1 1 2 1 ...
##  $ occupation   : Factor w/ 15 levels "?","Adm-clerical",..: 2 5 7 7 11 5 9 5 11 5 ...
##  $ relationship : Factor w/ 6 levels "Husband","Not-in-family",..: 2 1 2 1 6 6 2 1 2 1 ...
##  $ race         : Factor w/ 5 levels "Amer-Indian-Eskimo",..: 5 5 5 3 3 5 3 5 5 5 ...
##  $ sex          : Factor w/ 2 levels "Female","Male": 2 2 2 2 1 1 1 2 1 2 ...
##  $ capital_gain : int  2174 0 0 0 0 0 0 0 14084 5178 ...
##  $ capital_loss : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr_per_week  : int  40 13 40 40 40 40 16 45 50 40 ...
##  $ region       : Factor w/ 5 levels "Asia","Europe",..: 4 4 4 4 3 4 3 4 4 4 ...
##  $ income       : Factor w/ 2 levels "<=50K",">50K": 1 1 1 1 1 1 1 2 2 2 ...
```

Building a Model

```
library(caret)
```

```
## Loading required package: ggplot2
```

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

```
set.seed(2022)
index = createDataPartition(adult$income, p = 0.8, list = FALSE)
train = adult[index, ]
test = adult[-index, ]

model1 = glm(income ~., family = binomial(logit), data = train)
```

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

```
summary(model1)
```

```
##
## Call:
## glm(formula = income ~ ., family = binomial(logit), data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.1172  -0.5178  -0.1937   0.0000   3.6950
##
## Coefficients: (1 not defined because of singularities)
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -5.435e+00  4.042e-01 -13.446  < 2e-16 ***
## age                          2.647e-02  1.861e-03  14.222  < 2e-16 ***
## type_employerPrivate        -5.310e-01  1.052e-01  -5.049 4.43e-07 ***
## type_employerSelf-emp       -7.834e-01  1.166e-01  -6.720 1.82e-11 ***
## type_employerSL-gov         -7.758e-01  1.184e-01  -6.554 5.59e-11 ***
## type_employerUnemployed     -1.300e+01  2.394e+02  -0.054 0.956698
## fnlwgt                       7.038e-07  1.935e-07   3.638 0.000275 ***
## education11th               -1.698e-02  2.349e-01  -0.072 0.942379
## education12th                1.330e-01  3.193e-01   0.416 0.677078
## education1st-4th            -6.743e-01  5.334e-01  -1.264 0.206124
## education5th-6th            -4.426e-01  3.774e-01  -1.173 0.240891
## education7th-8th            -6.481e-01  2.622e-01  -2.472 0.013447 *
## education9th                -3.197e-01  2.854e-01  -1.120 0.262710
## educationAssoc-acdm          1.180e+00  1.949e-01   6.054 1.41e-09 ***
## educationAssoc-voc           1.128e+00  1.874e-01   6.020 1.75e-09 ***
## educationBachelors           1.772e+00  1.739e-01  10.192  < 2e-16 ***
## educationDoctorate           2.835e+00  2.389e-01  11.864  < 2e-16 ***
## educationHS-grad             6.174e-01  1.692e-01   3.649 0.000263 ***
## educationMasters             2.142e+00  1.863e-01  11.499  < 2e-16 ***
## educationPreschool          -1.781e+01  1.003e+02  -0.178 0.858984
## educationProf-school         2.616e+00  2.239e-01  11.685  < 2e-16 ***
```

```
## educationSome-college         9.926e-01  1.717e-01   5.781 7.43e-09 ***
## education_num                        NA         NA      NA       NA    
## maritalNever-married         -1.278e+00  1.883e-01  -6.789 1.13e-11 ***
## maritalNot-Married           -7.802e-01  1.874e-01  -4.163 3.14e-05 ***
## occupationArmed-Forces       -1.216e+01  3.728e+02  -0.033 0.973990    
## occupationCraft-repair        1.096e-01  8.949e-02   1.224 0.220840    
## occupationExec-managerial     8.242e-01  8.645e-02   9.534  < 2e-16 ***
## occupationFarming-fishing    -1.181e+00  1.593e-01  -7.413 1.24e-13 ***
## occupationHandlers-cleaners  -7.035e-01  1.608e-01  -4.376 1.21e-05 ***
## occupationMachine-op-inspct  -3.197e-01  1.143e-01  -2.796 0.005170 ** 
## occupationOther-service      -7.590e-01  1.302e-01  -5.830 5.53e-09 ***
## occupationPriv-house-serv    -3.787e+00  1.828e+00  -2.071 0.038337 *  
## occupationProf-specialty      5.414e-01  9.172e-02   5.902 3.59e-09 ***
## occupationProtective-serv     5.938e-01  1.415e-01   4.195 2.73e-05 ***
## occupationSales               3.301e-01  9.226e-02   3.578 0.000347 ***
## occupationTech-support        6.884e-01  1.233e-01   5.584 2.35e-08 ***
## occupationTransport-moving   -1.109e-01  1.127e-01  -0.984 0.325013    
## relationshipNot-in-family    -8.689e-01  1.845e-01  -4.709 2.49e-06 ***
## relationshipOther-relative   -1.250e+00  2.589e-01  -4.829 1.37e-06 ***
## relationshipOwn-child        -1.877e+00  2.317e-01  -8.102 5.40e-16 ***
## relationshipUnmarried        -9.009e-01  2.056e-01  -4.383 1.17e-05 ***
## relationshipWife              1.404e+00  1.165e-01  12.049  < 2e-16 ***
## raceAsian-Pac-Islander        6.055e-01  2.978e-01   2.033 0.042047 *  
## raceBlack                     3.927e-01  2.624e-01   1.496 0.134538    
## raceOther                    -3.554e-02  3.971e-01  -0.089 0.928687    
## raceWhite                     5.789e-01  2.501e-01   2.315 0.020612 *  
## sexMale                       8.968e-01  8.909e-02  10.066  < 2e-16 ***
## capital_gain                  3.211e-04  1.176e-05  27.306  < 2e-16 ***
## capital_loss                  6.311e-04  4.238e-05  14.892  < 2e-16 ***
## hr_per_week                   3.109e-02  1.879e-03  16.552  < 2e-16 ***
## regionEurope                  1.461e-01  2.419e-01   0.604 0.545931    
## regionLatin.and.South.America -3.588e-01  2.428e-01  -1.478 0.139534    
## regionNorth.America           1.806e-01  1.954e-01   0.924 0.355391    
## regionOther                  -4.006e-01  2.185e-01  -1.834 0.066690 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 27586  on 24574  degrees of freedom
## Residual deviance: 15944  on 24521  degrees of freedom
## AIC: 16052
## 
## Number of Fisher Scoring iterations: 13
```

Based on the first logistic regression model, the predictors with the most amount of significance in predicting one's income is age, employment type, completion of higher education, never being married, most occupations (executive manager, farming/fishing, handlers/cleaners, machine operator, professor, protective service, sales, and tech support), relationship status (not in a family, having other relatives, having a child, unmarried, and having a wife), being a male, capital gain/loss, and number of hours worked per week. All of these factors play a strong role in predicting someone's income. The rest of the predictors do not play a significant role in predicitng income.

In order to make a more accurate model, I will create a second model using step-wise regression by selecting the more relevant predictors for income.

```r
model2 = step(model1)
```

```
## Start:  AIC=16052.11
## income ~ age + type_employer + fnlwgt + education + education_num +
##     marital + occupation + relationship + race + sex + capital_gain +
##     capital_loss + hr_per_week + region
```

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

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

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

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

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

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

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

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

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

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

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

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

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

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

```
##
## Step:  AIC=16052.11
## income ~ age + type_employer + fnlwgt + education + marital +
##     occupation + relationship + race + sex + capital_gain + capital_loss +
##     hr_per_week + region
```

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

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

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

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

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

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

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

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

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

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

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

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

```
##                 Df Deviance   AIC
## <none>                15944 16052
## - race           4    15959 16059
## - fnlwgt         1    15957 16063
## - region         4    15972 16072
## - marital        2    16000 16104
## - type_employer  4    16007 16107
## - sex            1    16050 16156
## - age            1    16148 16254
## - capital_loss   1    16172 16278
## - hr_per_week    1    16227 16333
## - relationship   5    16240 16338
## - occupation    13    16468 16550
## - education     15    16761 16839
## - capital_gain   1    17393 17499
```

```
summary(model2)
```

```
##
## Call:
## glm(formula = income ~ age + type_employer + fnlwgt + education +
##     marital + occupation + relationship + race + sex + capital_gain +
##     capital_loss + hr_per_week + region, family = binomial(logit),
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.1172  -0.5178  -0.1937   0.0000   3.6950
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -5.435e+00  4.042e-01 -13.446  < 2e-16 ***
## age                           2.647e-02  1.861e-03  14.222  < 2e-16 ***
## type_employerPrivate         -5.310e-01  1.052e-01  -5.049 4.43e-07 ***
## type_employerSelf-emp        -7.834e-01  1.166e-01  -6.720 1.82e-11 ***
## type_employerSL-gov          -7.758e-01  1.184e-01  -6.554 5.59e-11 ***
## type_employerUnemployed      -1.300e+01  2.394e+02  -0.054 0.956698
## fnlwgt                        7.038e-07  1.935e-07   3.638 0.000275 ***
## education11th                -1.698e-02  2.349e-01  -0.072 0.942379
## education12th                 1.330e-01  3.193e-01   0.416 0.677078
```

```
## education1st-4th              -6.743e-01  5.334e-01  -1.264 0.206124
## education5th-6th              -4.426e-01  3.774e-01  -1.173 0.240891
## education7th-8th              -6.481e-01  2.622e-01  -2.472 0.013447 *
## education9th                  -3.197e-01  2.854e-01  -1.120 0.262710
## educationAssoc-acdm            1.180e+00  1.949e-01   6.054 1.41e-09 ***
## educationAssoc-voc             1.128e+00  1.874e-01   6.020 1.75e-09 ***
## educationBachelors             1.772e+00  1.739e-01  10.192  < 2e-16 ***
## educationDoctorate             2.835e+00  2.389e-01  11.864  < 2e-16 ***
## educationHS-grad               6.174e-01  1.692e-01   3.649 0.000263 ***
## educationMasters               2.142e+00  1.863e-01  11.499  < 2e-16 ***
## educationPreschool            -1.781e+01  1.003e+02  -0.178 0.858984
## educationProf-school           2.616e+00  2.239e-01  11.685  < 2e-16 ***
## educationSome-college          9.926e-01  1.717e-01   5.781 7.43e-09 ***
## maritalNever-married          -1.278e+00  1.883e-01  -6.789 1.13e-11 ***
## maritalNot-Married            -7.802e-01  1.874e-01  -4.163 3.14e-05 ***
## occupationArmed-Forces        -1.216e+01  3.728e+02  -0.033 0.973990
## occupationCraft-repair         1.096e-01  8.949e-02   1.224 0.220840
## occupationExec-managerial      8.242e-01  8.645e-02   9.534  < 2e-16 ***
## occupationFarming-fishing     -1.181e+00  1.593e-01  -7.413 1.24e-13 ***
## occupationHandlers-cleaners   -7.035e-01  1.608e-01  -4.376 1.21e-05 ***
## occupationMachine-op-inspct   -3.197e-01  1.143e-01  -2.796 0.005170 **
## occupationOther-service       -7.590e-01  1.302e-01  -5.830 5.53e-09 ***
## occupationPriv-house-serv     -3.787e+00  1.828e+00  -2.071 0.038337 *
## occupationProf-specialty       5.414e-01  9.172e-02   5.902 3.59e-09 ***
## occupationProtective-serv      5.938e-01  1.415e-01   4.195 2.73e-05 ***
## occupationSales                3.301e-01  9.226e-02   3.578 0.000347 ***
## occupationTech-support         6.884e-01  1.233e-01   5.584 2.35e-08 ***
## occupationTransport-moving    -1.109e-01  1.127e-01  -0.984 0.325013
## relationshipNot-in-family     -8.689e-01  1.845e-01  -4.709 2.49e-06 ***
## relationshipOther-relative    -1.250e+00  2.589e-01  -4.829 1.37e-06 ***
## relationshipOwn-child         -1.877e+00  2.317e-01  -8.102 5.40e-16 ***
## relationshipUnmarried         -9.009e-01  2.056e-01  -4.383 1.17e-05 ***
## relationshipWife               1.404e+00  1.165e-01  12.049  < 2e-16 ***
## raceAsian-Pac-Islander         6.055e-01  2.978e-01   2.033 0.042047 *
## raceBlack                      3.927e-01  2.624e-01   1.496 0.134538
## raceOther                     -3.554e-02  3.971e-01  -0.089 0.928687
## raceWhite                      5.789e-01  2.501e-01   2.315 0.020612 *
## sexMale                        8.968e-01  8.909e-02  10.066  < 2e-16 ***
## capital_gain                   3.211e-04  1.176e-05  27.306  < 2e-16 ***
## capital_loss                   6.311e-04  4.238e-05  14.892  < 2e-16 ***
## hr_per_week                    3.109e-02  1.879e-03  16.552  < 2e-16 ***
## regionEurope                   1.461e-01  2.419e-01   0.604 0.545931
## regionLatin.and.South.America -3.588e-01  2.428e-01  -1.478 0.139534
## regionNorth.America            1.806e-01  1.954e-01   0.924 0.355391
## regionOther                   -4.006e-01  2.185e-01  -1.834 0.066690 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 27586  on 24574  degrees of freedom
## Residual deviance: 15944  on 24521  degrees of freedom
## AIC: 16052
##
```

```
## Number of Fisher Scoring iterations: 13
```

After running a second model with step-wise regression, it seems that the most significant predictors remained. Running this step=wise regression model did not appear to improve or worsen the previous model.

Now I will build a confusion matrix to compute accuracy, misclassification rate, recall, and precision.

```
test$predict = predict(object = model1, newdata = test, type = 'response')
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```
table(test$income, test$predict > 0.5)
```

```
##
##         FALSE TRUE
##   <=50K  4286  327
##   >50K    650  880
```

The accuracy of this model is $\frac{4286+880}{4286+880+650+327} = 0.8409572$
The misclassification rate of this model is 1 - accuracy: $1 - 0.8409572 = 0.1590428$
The recall of this model is $\frac{4286}{4286+327} = 0.9291134$
The precision of this model is $\frac{4286}{4286+650} = 0.8683144$

Overall, this is a good model in predicting whether person makes below or above $50,000, but the accuracy can be higher. I think one way this can be achieved is by entirely removing the predictors that have little to no significance in predicting income.