"Classifying income as over/under $50k per year"

Craig Haile

September 1, 2019

In partial fulfillment of the requirements of HarvardX: PH125.9x Data Science Capstone

Introduction

The purpose of this project is to create several classifcation models for data related to individual income level. In particular, we will consider numerical and categorical predictors for the outcome variable "income", which is binary with levels under/over $50000. We will use the data set "adult.csv". We derive three common classification models: Binary Logistic Regression through the glm function, K nearest neighbors through the knn function, and Random Forest. The Random Forest produces the greatest overall accuracy while the glm has the highest area under the ROC curve.

This is an extraction of 32,561 responses from the 1994 US Census data taken from the Kaggle list of curated datasets at https://www.kaggle.com/uciml/adult-census-income.

Exploratory Analysis

Looking at a selection of rows and columns of the dataset gives a feel for it's form.

```
## # A tibble: 6 x 15
##     age workclass fnlwgt education education.num marital.status occupation
##   <dbl> <chr>      <dbl> <chr>             <dbl> <chr>          <chr>
## 1    90 ?          77053 HS-grad               9 Widowed        ?
## 2    82 Private   132870 HS-grad               9 Widowed        Exec-mana~
## 3    66 ?         186061 Some-col~            10 Widowed        ?
## 4    54 Private   140359 7th-8th               4 Divorced       Machine-o~
## 5    41 Private   264663 Some-col~            10 Separated      Prof-spec~
## 6    34 Private   216864 HS-grad               9 Divorced       Other-ser~
## # ... with 8 more variables: relationship <chr>, race <chr>, sex <chr>,
## #   capital.gain <dbl>, capital.loss <dbl>, hours.per.week <dbl>,
## #   native.country <chr>, income <chr>
```

The Kaggle website only gives the levels of the categorical variables or if numerical describes them as continuous. However, we can reasonably infer their meaning as described below:

1. age: numeric, age of the respondent.
2. workclass: categorical, type of employment.
3. fnlwgt: numeric, reflects the number of people in the population with the same attributes as the respondent entry.
4. education: categorical, education level.
5. education.num: numeric, education level.
6. marital.status: categorical, marital status.
7. occupation: categorical, work occupation.
8. relationship: categorical, reflects whether the individual has a familial relationship relative to another person in the household.
9. race: categorical, description of race.
10. sex: categorical, description of sex as male or female.

11. capital.gain: numeric, reported capital gain in dollars.
12. capital.loss: numeric, reported capital loss in dollars.
13. hours.per.week: numeric, number of hours worked per week.
14. native.country: categorical, country of origin.
15. income: categorical binary, less or equal to \$50000 or greater than \$50000.

The primary goal is to predict income (>50K or <=50K) using the other variables as predictors. Looking at the distribution of incomes in the dataset,

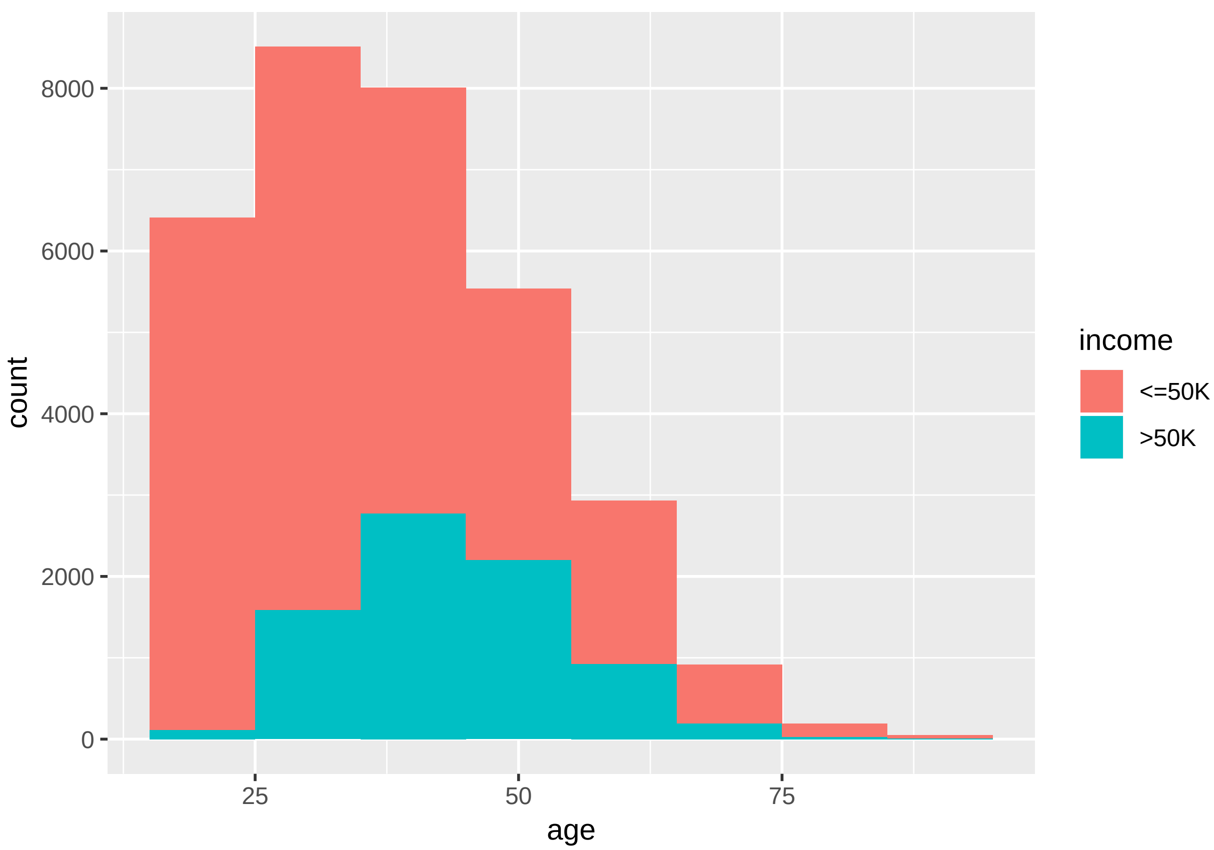
```
## # A tibble: 2 x 2
##   income count
##   <chr>  <int>
## 1 <=50K  24720
## 2 >50K    7841
```

it looks that about 3/4 of respondents had income under \$50000, while 1/4 had income over this mark. So from a naive perspective we could always just guess that someone had under \$50K income and we would have a 75% overall accuracy. We will try to beat that.

Identifying important predictors

To find out which of the predictor variables are most helpful in classification, we will construct some basic tables and visualizations of the predictor variables with respect to income.

For numeric variables we will construct histograms that indicate the distribution of the predictive variable along with the distribution of income. For categorical variables we provide a summary two-way table indicating the count in each income group for each level of the variable.



This graph shows that "middle" ages are the highest proportion of workers and the most likely to have >50K income.

```
##
##                    <=50K  >50K
##   ?                 1645   191
##   Federal-gov        589   371
##   Local-gov         1476   617
##   Never-worked         7     0
##   Private          17733  4963
##   Self-emp-inc       494   622
##   Self-emp-not-inc  1817   724
##   State-gov          945   353
##   Without-pay         14     0
```

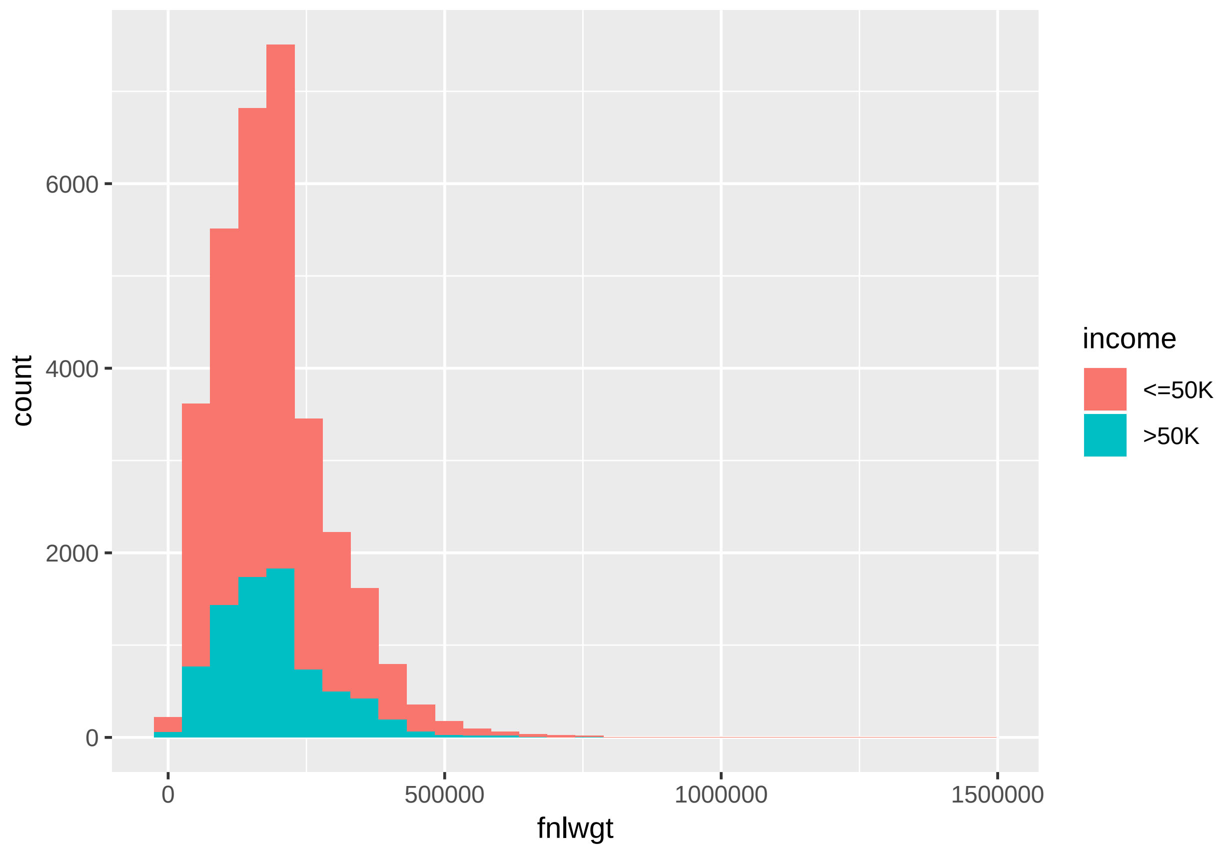
We note in this table that there are missing data, denoted with "?", and some very small levels such as "never-worked" or worked "without-pay". We will collapse these into a single level "other".

```
##
##                    <=50K  >50K
##   other             1666   191
##   Federal-gov        589   371
##   Local-gov         1476   617
##   Private          17733  4963
##   Self-emp-inc       494   622
##   Self-emp-not-inc  1817   724
##   State-gov          945   353
```

This leaves us with a reasonable number of levels (seven) that all have a significant number of values relative to the size of the dataset.

The graph of "fnlwgt" vs "income" shows the proportion of >50K to be fairly consistent for all the values, suggesting that this may not have much predictive value.

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



Looking at income relative to education

```
##
##                 <=50K >50K
##   10th            871   62
##   11th           1115   60
##   12th            400   33
##   1st-4th         162    6
##   5th-6th         317   16
##   7th-8th         606   40
##   9th             487   27
##   Assoc-acdm      802  265
##   Assoc-voc      1021  361
##   Bachelors      3134 2221
##   Doctorate       107  306
##   HS-grad        8826 1675
##   Masters         764  959
##   Preschool        51    0
##   Prof-school     153  423
##   Some-college   5904 1387
```

we see that generally more education leads to a higher proportion of respondents making >50K. This is seen more clearly by looking at the education.num variable, which converts the education levels to an numerical value, with higher values corresponding to a greater level of education.

```
##
##       <=50K >50K
##   1      51    0
```

```
##   2    162    6
##   3    317   16
##   4    606   40
##   5    487   27
##   6    871   62
##   7   1115   60
##   8    400   33
##   9   8826 1675
##   10  5904 1387
##   11  1021  361
##   12   802  265
##   13  3134 2221
##   14   764  959
##   15   153  423
##   16   107  306
```

Although there are many levels, because we can treat this as a numerical variable we will keep all.

Next we consider marital status.

```
##
##                         <=50K  >50K
##   Divorced               3980   463
##   Married-AF-spouse        13    10
##   Married-civ-spouse     8284  6692
##   Married-spouse-absent   384    34
##   Never-married         10192   491
##   Separated               959    66
##   Widowed                 908    85
```

Here again, with seven levels and some small counts we will do some combining. In particular the table seems to indicate that married (with spouse present) has a much higher proportion of income >50K than any category with individuals living alone, so we will collapse to two categories, married_together and not_together.

```
##
##                    <=50K  >50K
##   not_together     16423  1139
##   married_together  8297  6702
```

We face the same problem with occupation, many levels and some small counts,

```
##
##                     <=50K >50K
##   ?                  1652  191
##   Adm-clerical       3263  507
##   Armed-Forces          8    1
##   Craft-repair       3170  929
##   Exec-managerial    2098 1968
##   Farming-fishing     879  115
##   Handlers-cleaners  1284   86
##   Machine-op-inspct  1752  250
##   Other-service      3158  137
##   Priv-house-serv     148    1
##   Prof-specialty     2281 1859
##   Protective-serv     438  211
##   Sales              2667  983
```

```
##   Tech-support        645  283
##   Transport-moving   1277  320
```

and so we will again combine fields. There is certainly as much art as science in defining the new fields, but we perceive benefit in striving for a simpler model. We will call our fields Blue_Collar (Craft-repair,Farming-fishing,Handlers-cleaners,Machine-op-inspct,Transport-moving, White_Collar (Adm-clerical,Sales,Tech-support,Protective-serv), Exec_mgr_prof (Exec-managerial,Prof-specialty), and Service_other (?, Armed-Forces, Other-service, Priv-house-serv)

```
##
##                  <=50K >50K
##   Service_other  4966  330
##   White_Collar   7013 1984
##   Blue_Collar    8362 1700
##   Exec_mgr_prof  4379 3827
```

In the next table we look at the variable relationship. Considering how the proportions of >50K are considerably weighted toward those who are husbands and wives, it would seem this is redundant to marital status.

```
##
##                   <=50K >50K
##   Husband          7275 5918
##   Not-in-family    7449  856
##   Other-relative    944   37
##   Own-child        5001   67
##   Unmarried        3228  218
##   Wife              823  745
```

Now we consider race and sex. We say that race is largely white and sex is largely male, which may limit predictive value. Although there are some race categories that have fairly small counts, there are only five levels overall so we will not combine.

```
##
##                         <=50K  >50K
##   Amer-Indian-Eskimo     275    36
##   Asian-Pac-Islander     763   276
##   Black                 2737   387
##   Other                  246    25
##   White                20699  7117
```

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

The next variables considered together are capital gain and loss. Some summary tables of descriptive statistics show that while there is a wide range in dollar values (especially for capital gains), most respondents had value zero.

```r
summary(adult$capital.gain)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0       0       0    1078       0   99999
```

```r
sum(adult$capital.gain==0)/length(adult$capital.gain)
```

```
## [1] 0.9167102
```

```r
summary(adult$capital.loss)
```

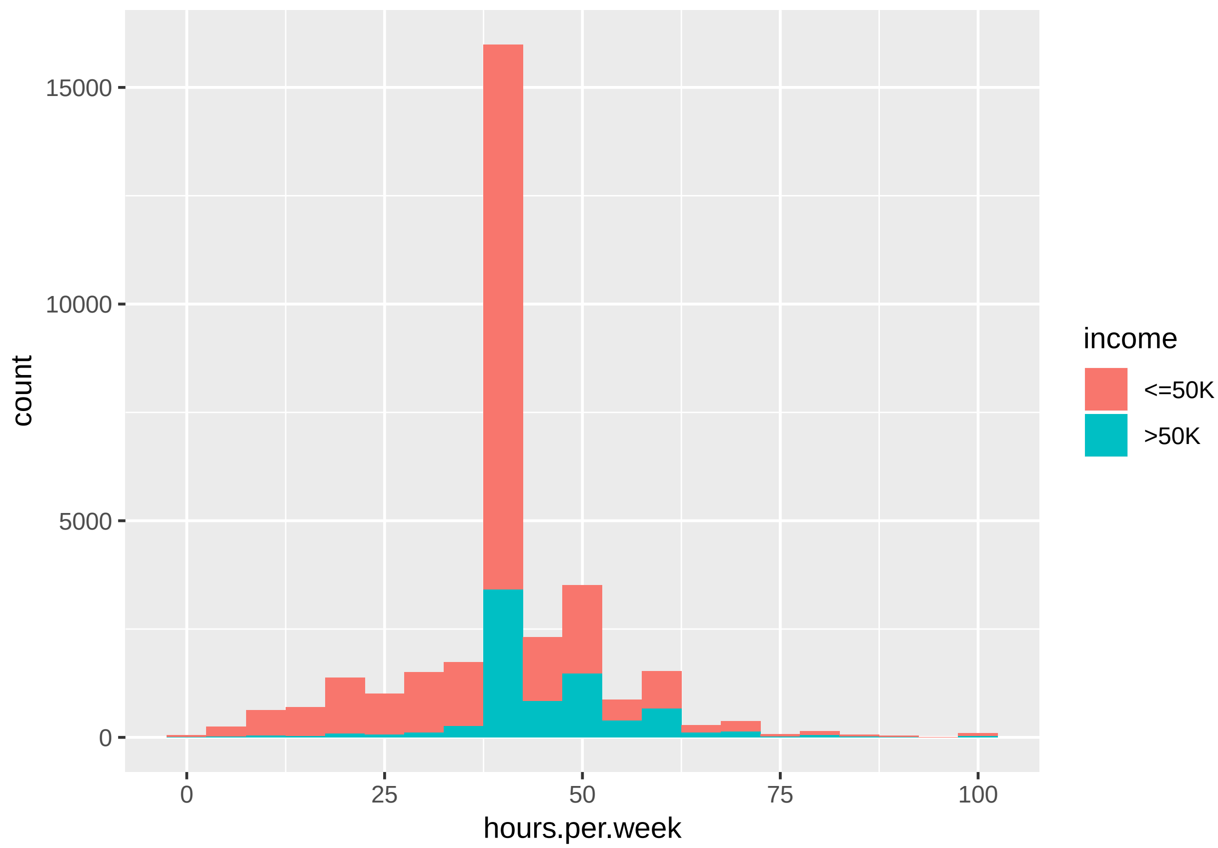
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0     0.0    87.3     0.0  4356.0
```

```r
sum(adult$capital.loss==0)/length(adult$capital.loss)
```

```
## [1] 0.9533491
```

As a matter of fact, we see 92% and 95% of capital gains and losses, respectively, have zero values, making these variables that would likely have little predictive value.

A histogram of hours worked per week



indicates (to no surprise) that most work around 40 hours per week, and few people working less than 40 hours earn more than 50K.

Finally, we will look at the native country of the respondent.

```
##
##                              <=50K  >50K
##   ?                            437   146
##   Cambodia                      12     7
##   Canada                        82    39
##   China                         55    20
##   Columbia                      57     2
##   Cuba                          70    25
##   Dominican-Republic            68     2
##   Ecuador                       24     4
```

```
##   El-Salvador                     97     9
##   England                         60    30
##   France                          17    12
##   Germany                         93    44
##   Greece                          21     8
##   Guatemala                       61     3
##   Haiti                           40     4
##   Holand-Netherlands               1     0
##   Honduras                        12     1
##   Hong                            14     6
##   Hungary                         10     3
##   India                           60    40
##   Iran                            25    18
##   Ireland                         19     5
##   Italy                           48    25
##   Jamaica                         71    10
##   Japan                           38    24
##   Laos                            16     2
##   Mexico                         610    33
##   Nicaragua                       32     2
##   Outlying-US(Guam-USVI-etc)      14     0
##   Peru                            29     2
##   Philippines                    137    61
##   Poland                          48    12
##   Portugal                        33     4
##   Puerto-Rico                    102    12
##   Scotland                         9     3
##   South                           64    16
##   Taiwan                          31    20
##   Thailand                        15     3
##   Trinadad&Tobago                 17     2
##   United-States                21999  7171
##   Vietnam                         62     5
##   Yugoslavia                      10     6
```

Because there is such a high proportion of those born in the United States and such a multitude of levels, we will (crudely) reduce to either born in the United States or born outside the United States.

```
##
##              <=50K  >50K
##   Outside_US  2721   670
##   US         21999  7171
```

Reduce Data set to important predictors

Now that our initial investigations are done we will reduce the variables in the dataset. In particular, we will eliminate fnlwgt, education (we will keep education.num instead), relationship (largely redundant with marital status), capital.gain and capital.loss (more than 90% zeros).

```
#reduce the dataset for variables considered in model
adult <- adult %>% select(age,workclass,education.num,marital.status,occupation,race,sex,hours.per.week
head(adult)
```

```
## # A tibble: 6 x 10
##     age workclass education.num marital.status occupation race  sex
```

```
##   <dbl> <fct>             <dbl> <fct>          <fct>      <chr> <chr>
## 1    90 other                 9 not_together   Service_o~ White Fema~
## 2    82 Private               9 not_together   Exec_mgr_~ White Fema~
## 3    66 other                10 not_together   Service_o~ Black Fema~
## 4    54 Private               4 not_together   Blue_Coll~ White Fema~
## 5    41 Private              10 not_together   Exec_mgr_~ White Fema~
## 6    34 Private               9 not_together   Service_o~ White Fema~
## # ... with 3 more variables: hours.per.week <dbl>, native.country <chr>,
## #   income <chr>
```

We convert income into a binary variable with income less than \$50000 assigned a zero and greater than or equal to \$50000 assigned one for use in some graphs.

Finally, we split the data into a training and validation sets. 75% of the data is used in the training set and 25% is reserved for validation (testing).

Models and Analysis

Classification Model: General Linear Model (glm)

The first classification model considered will be a binary logistic regression model using glm. We will use all predictors in our reduced dataset.

```
## GLM Binary Logistic Model

default_glm_mod = train(
  form = income ~ age+workclass+education.num+marital.status+occupation+race+sex+hours.per.week+native.c
  data = train_set,
  trControl = trainControl(method = "cv", number = 5),
  method = "glm",
  family = "binomial"
)

##GLM Model summary
summary(default_glm_mod)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7229  -0.5650  -0.2481  -0.0637   3.5069
##
## Coefficients:
##                                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)                    -9.753818   0.301790 -32.320  < 2e-16 ***
## age                             0.030388   0.001628  18.664  < 2e-16 ***
## `workclassFederal-gov`          0.104854   0.178561   0.587 0.557059
## `workclassLocal-gov`           -0.507600   0.164188  -3.092 0.001991 **
## workclassPrivate               -0.346773   0.151023  -2.296 0.021667 *
## `workclassSelf-emp-inc`        -0.003932   0.173555  -0.023 0.981924
## `workclassSelf-emp-not-inc`    -0.807205   0.161575  -4.996 5.86e-07 ***
## `workclassState-gov`           -0.717208   0.175997  -4.075 4.60e-05 ***
```

```
## education.num                    0.285735   0.009665  29.563  < 2e-16 ***
## marital.statusmarried_together  2.332582   0.049935  46.712  < 2e-16 ***
## occupationWhite_Collar          1.211720   0.115528  10.489  < 2e-16 ***
## occupationBlue_Collar           0.683571   0.115651   5.911 3.41e-09 ***
## occupationExec_mgr_prof         1.657993   0.116557  14.225  < 2e-16 ***
## `raceAsian-Pac-Islander`        0.425573   0.255842   1.663 0.096228 .
## raceBlack                       0.375749   0.239052   1.572 0.115991
## raceOther                      -0.212787   0.383202  -0.555 0.578698
## raceWhite                       0.466469   0.227664   2.049 0.040469 *
## sexMale                         0.213727   0.053586   3.988 6.65e-05 ***
## hours.per.week                  0.030619   0.001708  17.923  < 2e-16 ***
## native.countryUS                0.279360   0.075774   3.687 0.000227 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 26959  on 24419  degrees of freedom
## Residual deviance: 17465  on 24400  degrees of freedom
## AIC: 17505
##
## Number of Fisher Scoring iterations: 6
```

We see all the predictor variables are significant for at least some levels. Next we produce the confusion matrix to see how well our model does in classification for the test set.

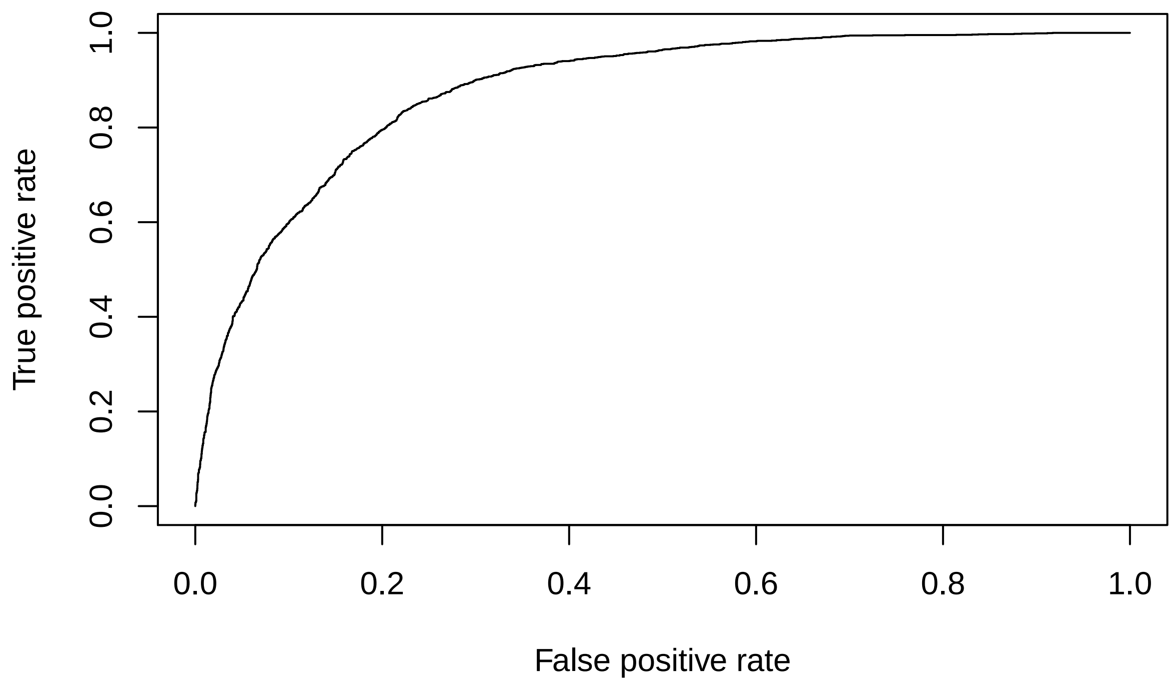
```
##confusion Matrix
y_hat_glm<-predict(default_glm_mod, newdata = test_set)

table(predicted=y_hat_glm,actual=test_set$income)
```

```
##          actual
## predicted <=50K >50K
##     <=50K  5723  913
##     >50K    457 1048
```

From the confusion matrix we compute standard accuracy and the F1 measure of accuracy.

```
##accuracy
calc_acc = function(actual, predicted) {
  mean(actual == predicted)
}

acc_glm<-calc_acc(actual = test_set$income,
         predicted = predict(default_glm_mod, newdata = test_set))

f1_glm<-F_meas(factor(y_hat_glm),factor(test_set$income))
```

and graph the ROC curve



K Nearest neighbors (KNN)

Next we turn to a knn model.

```
default_knn_mod = train(
  income ~ age+workclass+education.num+marital.status+occupation+race+sex+hours.per.week+native.country
  data = train_set,
  method = "knn",
  trControl = trainControl(method = "cv", number = 5),
  preProcess = c("center", "scale"),
  tuneGrid = expand.grid(k = seq(23, 25, by = 2))
)
```

We attempt to tune with various values of k, ultimately arriving at a best model of k = 23.

```
default_knn_mod$finalModel
```

```
## 25-nearest neighbor model
## Training set outcome distribution:
##
## <=50K  >50K
## 18540  5880
```

As before we compute accuracy

```
#knn accuracy
```

```
calc_acc = function(actual, predicted) {
  mean(actual == predicted)
}
acc_knn<-calc_acc(actual = test_set$income,
         predicted = predict(default_knn_mod, newdata = test_set))
acc_knn
```

```
## [1] 0.8283995
```

and the confusion matrix and F1 measure,

```
##confusion Matrix
y_hat_knn<-predict(default_knn_mod, newdata = test_set)

table(predicted=y_hat_knn,actual=test_set$income)
```

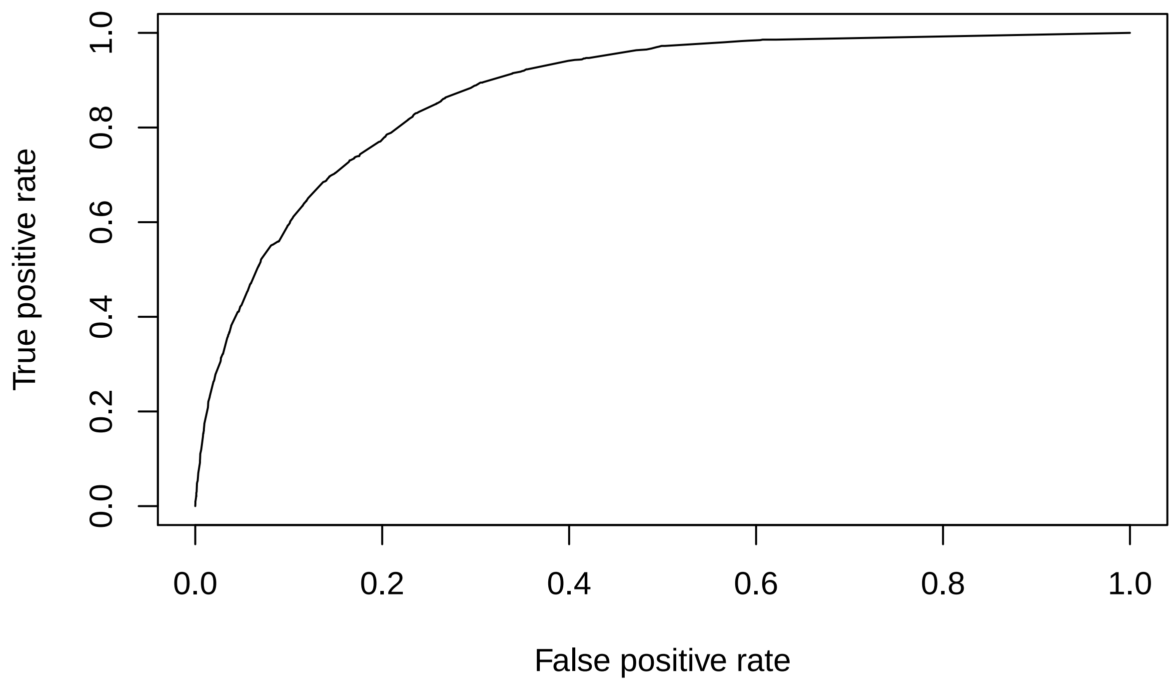
```
##          actual
## predicted <=50K >50K
##     <=50K  5649  873
##     >50K    531 1088
```

```
f1_knn<-F_meas(factor(y_hat_knn),factor(test_set$income))
f1_knn
```

```
## [1] 0.8894662
```

as well as the ROC curve.

```
## ROC Curve

## predict probabilities rather than binary
p_hat_knn<-predict(default_knn_mod, newdata = test_set, type = "prob")

pr2 <- prediction(p_hat_knn[2], test_set$income_b)
prf_knn <- performance(pr2, measure = "tpr", x.measure = "fpr")
plot(prf_knn)
```



We see a slightly lower overall accuracy and sensitivity with knn compared to glm, although a slight improvement in specificity.

Random Forest

Finally we consider a Random Forest model

```
rf <- randomForest(as.factor(income) ~ age+workclass+education.num+marital.status+occupation+hours.per.w
rf.pred.prob <- predict(rf, newdata = test_set, type = 'prob')
rf.pred <- predict(rf, newdata = test_set, type = 'class')
# confusion matrix
tb <- table(rf.pred, test_set$income)
tb
```

```
##
## rf.pred <=50K >50K
##   <=50K  5698  851
##   >50K    482 1110
```

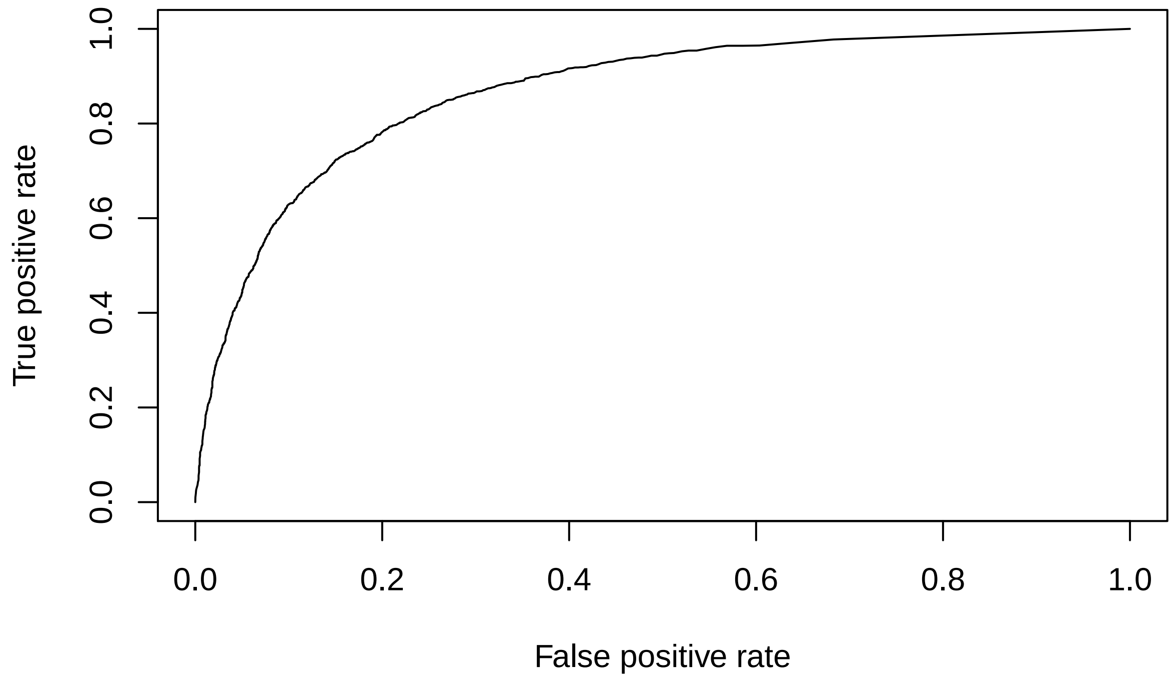
```
calc_acc = function(actual, predicted) {
  mean(actual == predicted)
}

acc_rf<-calc_acc(actual = test_set$income,
         predicted = predict(rf, newdata = test_set))
acc_rf
```

```
## [1] 0.8363837
```

```
f1_rf<-F_meas(rf.pred,factor(test_set$income))
f1_rf
```

```
## [1] 0.8952785
```

```
p_hat_rf<-as.data.frame(rf.pred.prob)
pr3 <- prediction(p_hat_rf[2], test_set$income_b)
prf_rf <- performance(pr3, measure = "tpr", x.measure = "fpr")
plot(prf_rf)
```



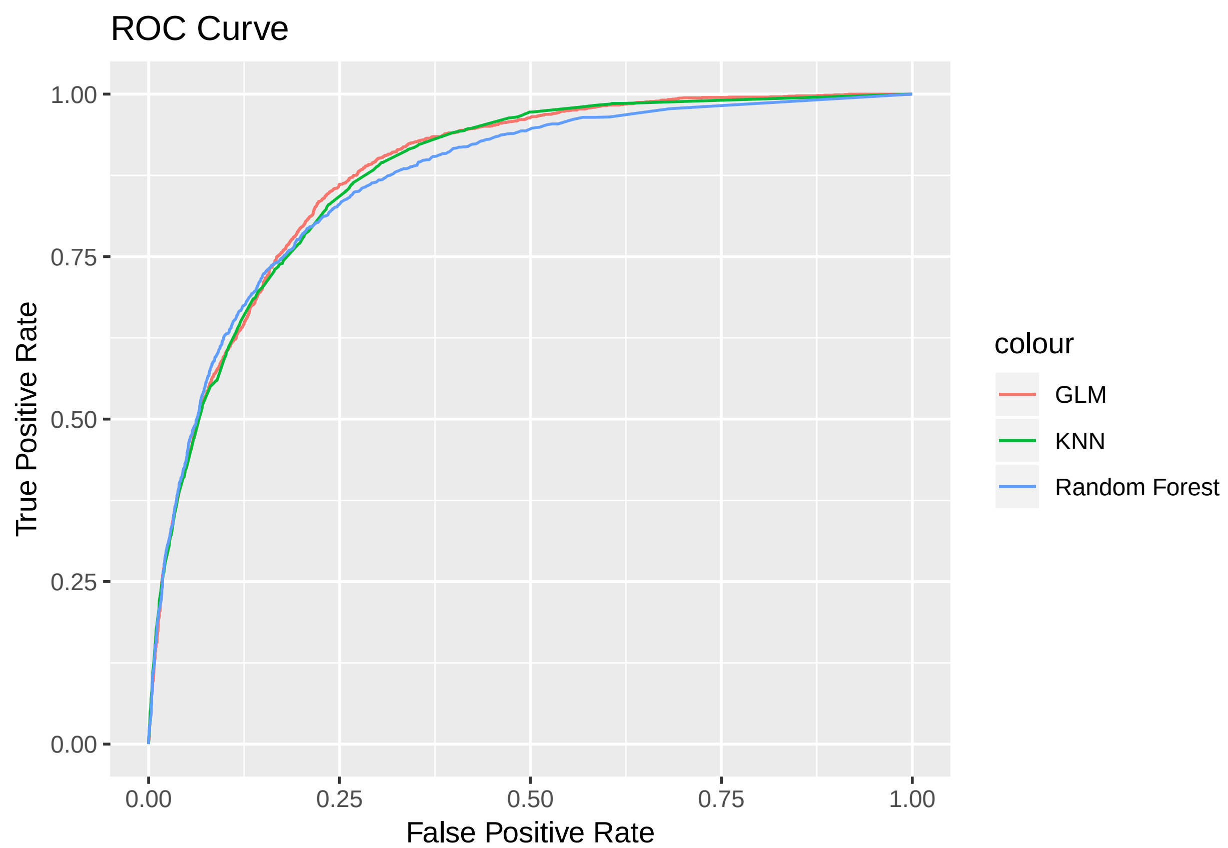
We see that this model has an improved overall accuracy and F1 score over both the previous models.

Summary of Results

We now summarize the results of the different models using overall accuracy and F1 score, the harmonic mean of precision and recall.

method	Accuracy	F1
GLM	0.8317160	0.8931024
KNN	0.8283995	0.8894662
Random Forest	0.8363837	0.8952785

Alternatively, we plot the ROC curves and compare the areas under the curve.




```
##                Area Under ROC Curve
## GLM                          0.8790
## KNN                          0.8760
## Random Forest                0.8676
```

Conclusion

We see we get conflicting results. By accuracy Random Forest > GLM > KNN, but by the ROC curve GLM > KNN > Random Forest. Ultimately one might choose the GLM results just for the ease of interpretation and understandability.

Limitations

We combined some features and didn't consider others that might have somewhat improved the overall accuracy but would have led to longer runtimes on computationally intensive techniques.