

Census Income Prediction Report

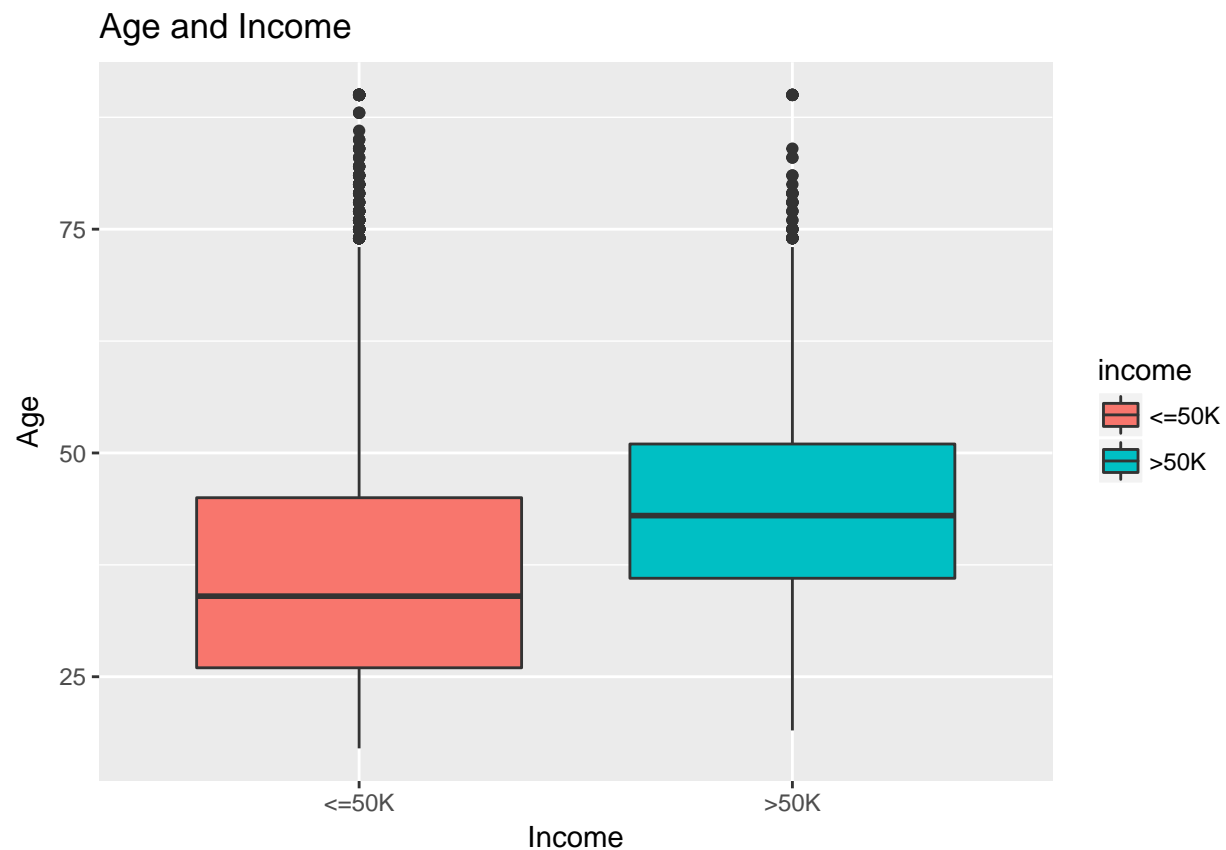
Jimmy Chan, Benny Chen

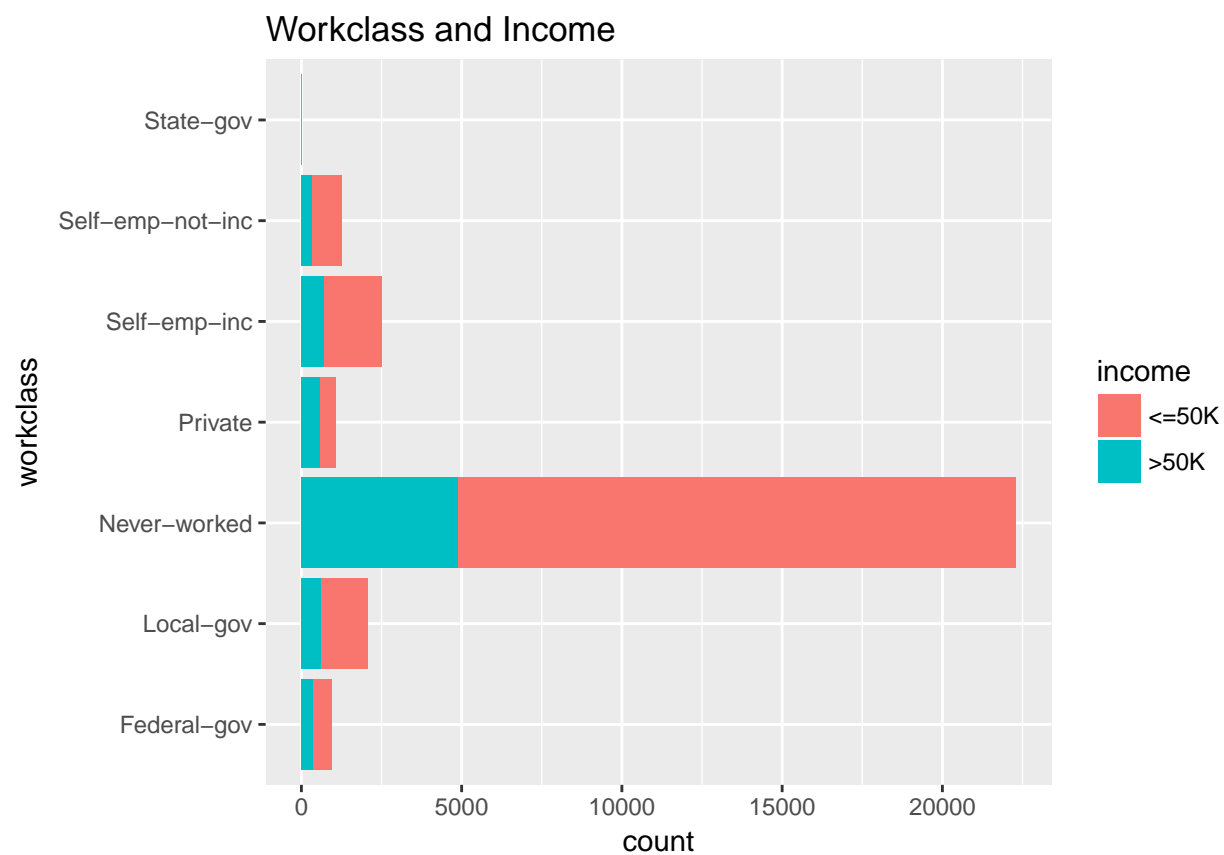
Dec 11, 2017

Introduction

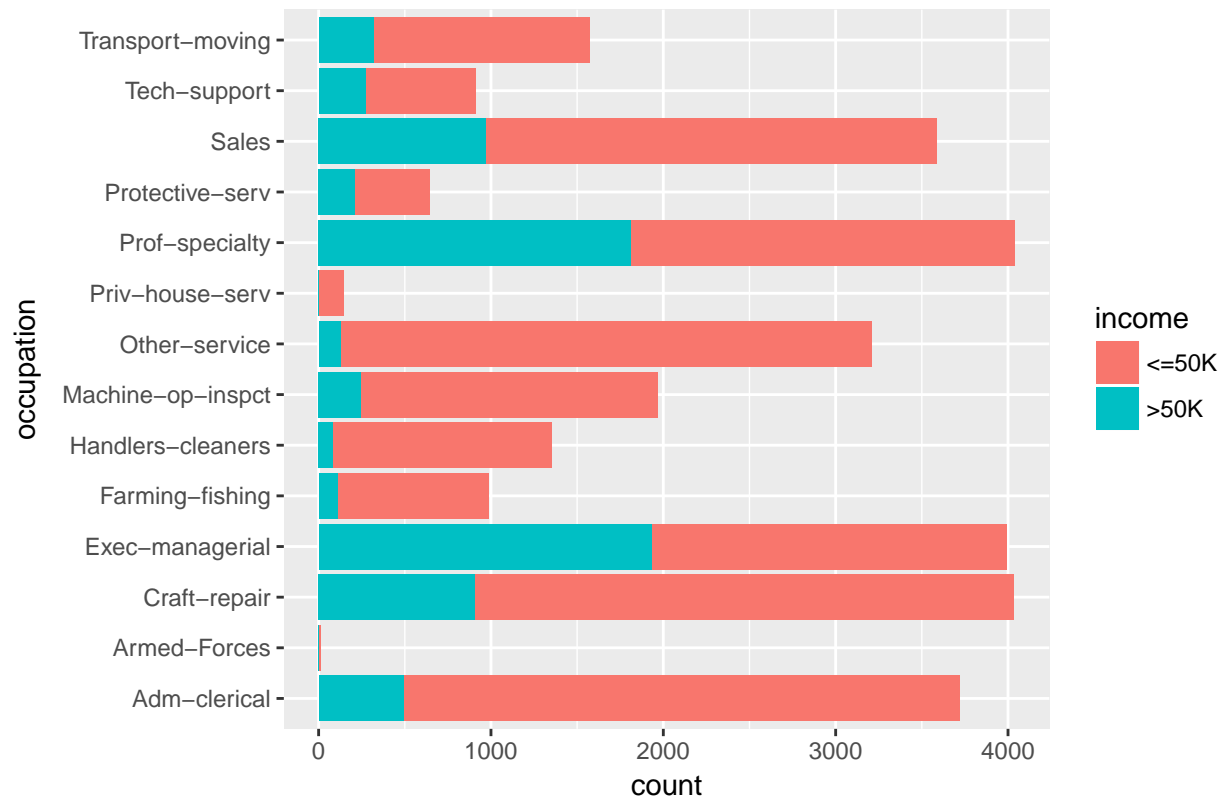
Using the data from the Census Income Dataset, we wanted to see which tree-based method had the highest predictive accuracy and which variables have the strongest predictive power in regards to whether an individual earns over \$50000 in income. We compare three methods - classification trees, bagged, trees, and random forest. We begin with preprocessing and exploration of the data. Then, we fit the three models using the training data. With the most accurate model, we fit the test data.

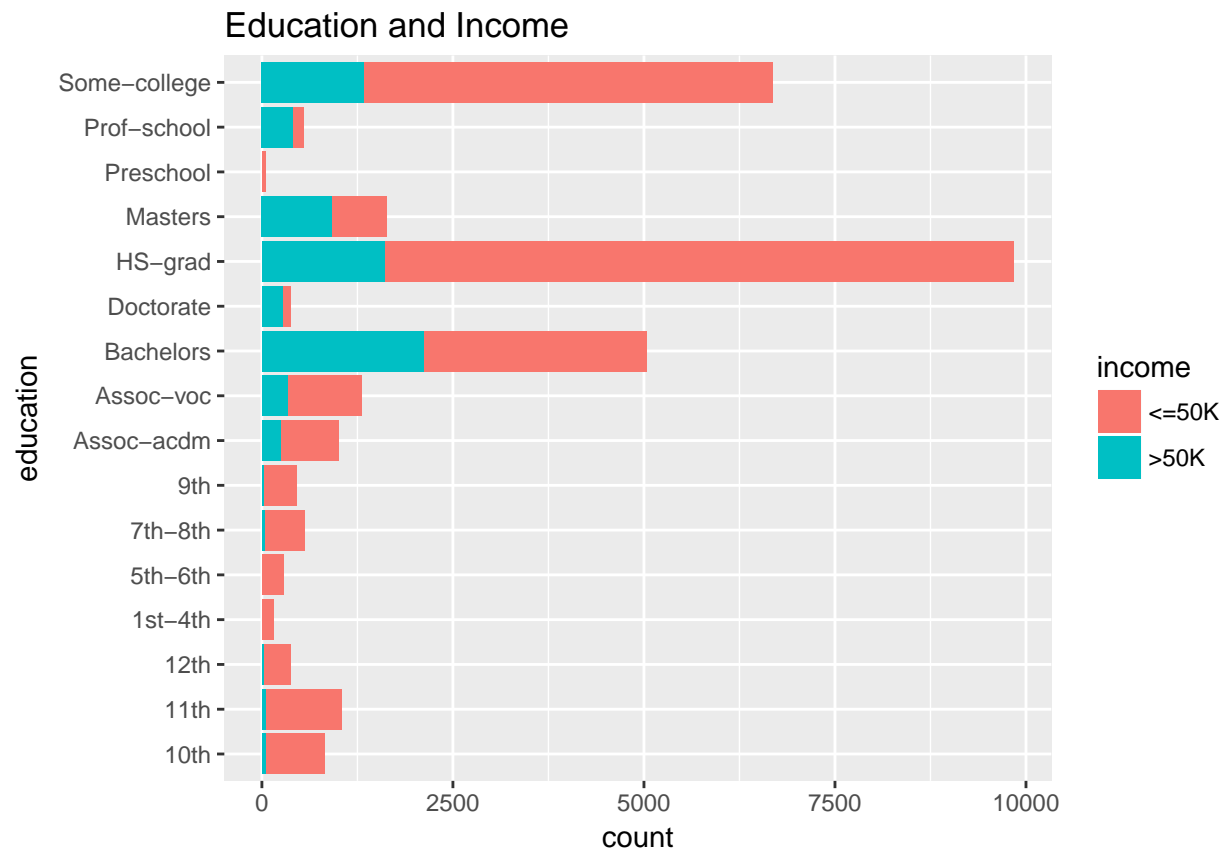
Exploratory Data Analysis

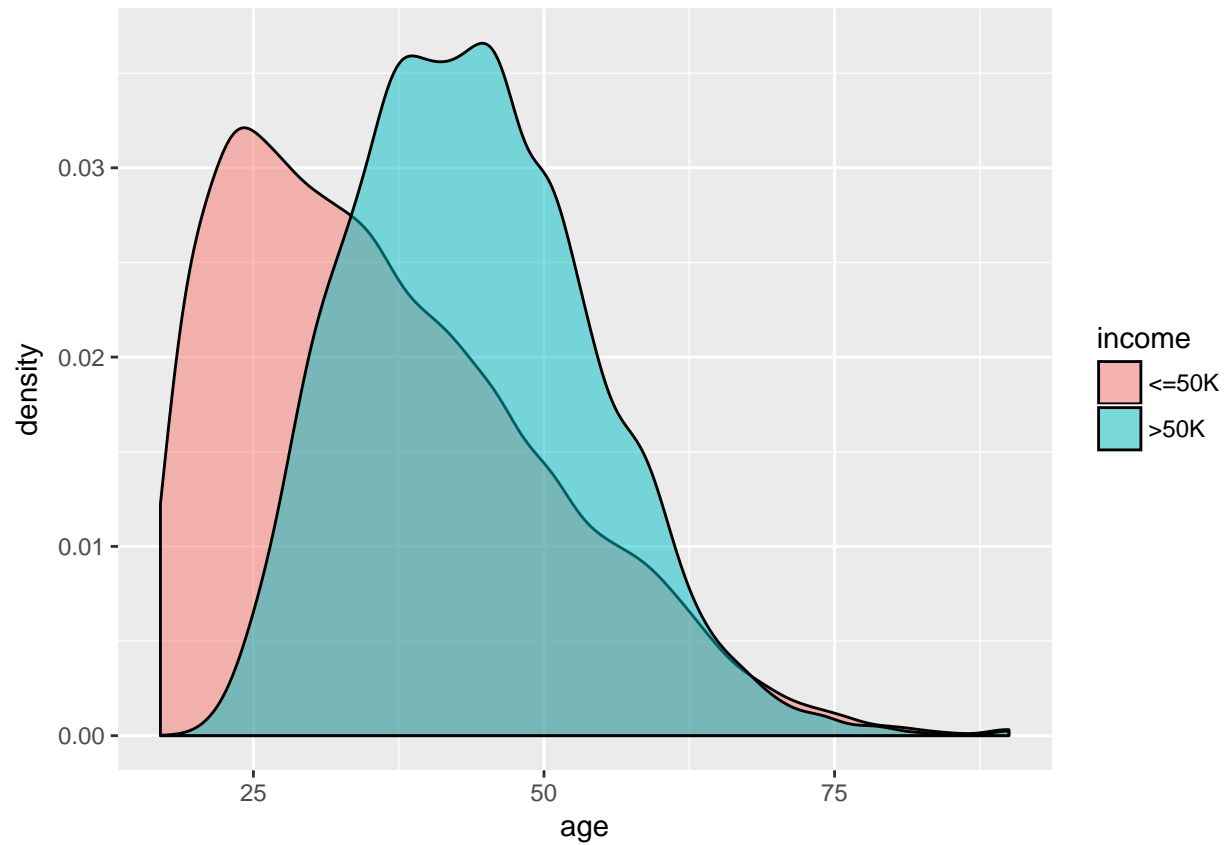




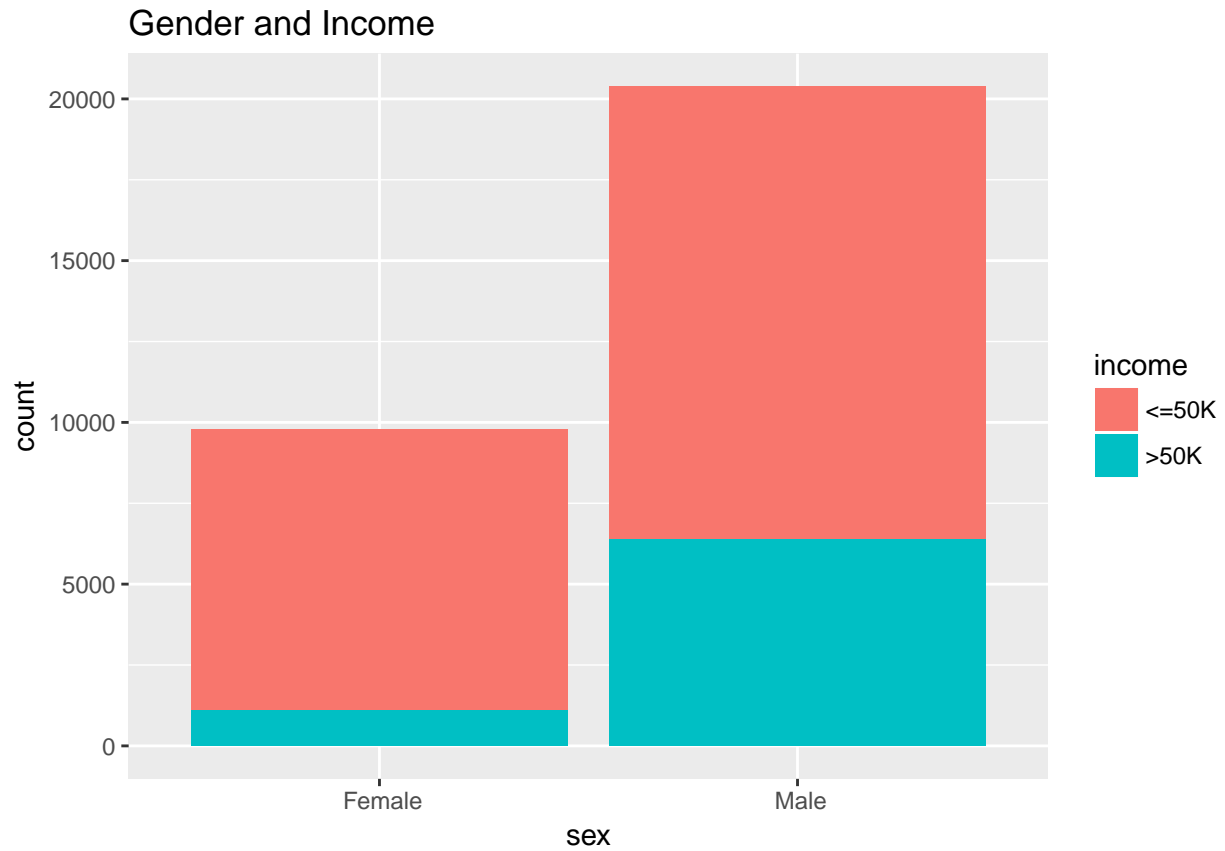
Occupation and Income



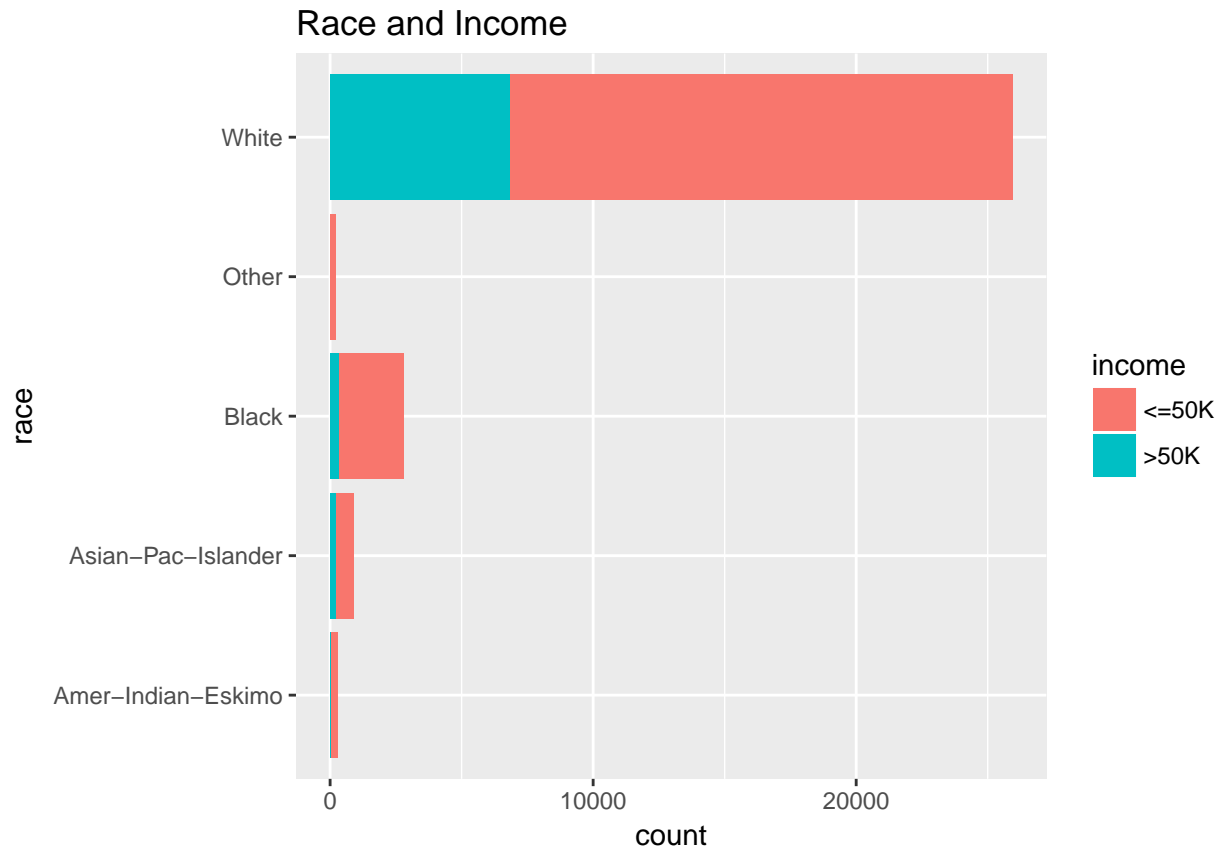




The median age of people making more than 50K is higher than the median age of people making less than 50k. Most people earning more than 50K are around 50 years old. This is consistent with the logic that older people tend to earn more.



The data has much more males than females so the barplot is not balanced. It appears that a people making less than 50K make up a larger portion of the female population than they do for males, although for the dataset, people making fewer than 50K make up the large majority of the data, so it is not too different from the overall sample.



From the barplot, it appears that a larger proportion of whites have an income greater than 50k compared to the proportion for the other races.

Data Processing

First, we needed to remove missing values that were found in the workclass, native_country, and occupation variables. Removal was appropriate as observations with these missing values made up about 2400 of the 32561 observations. To be used with the models, we needed to aggregate some of the countries of the native_country variable. We chose to aggregate the countries with the 10 fewest occurrences in the data into the “other” label, which totaled 137 observations. Based on our prior knowledge, we believed that age, education, and occupation would be important variables. Looking at the proportion of income levels, age and education appear the most different for the two groups. For many of the variables, initial plots were not too insightful due to the fact that the data was not very well balanced. We noticed that the training data had a much larger proportion of incomes greater than \$50K, so the AUC metric would be useful.

Our last step for this phase was to remove some of the variables. We felt that “education” and “education_num” variables were very well correlated so we would only use one to fit our models.

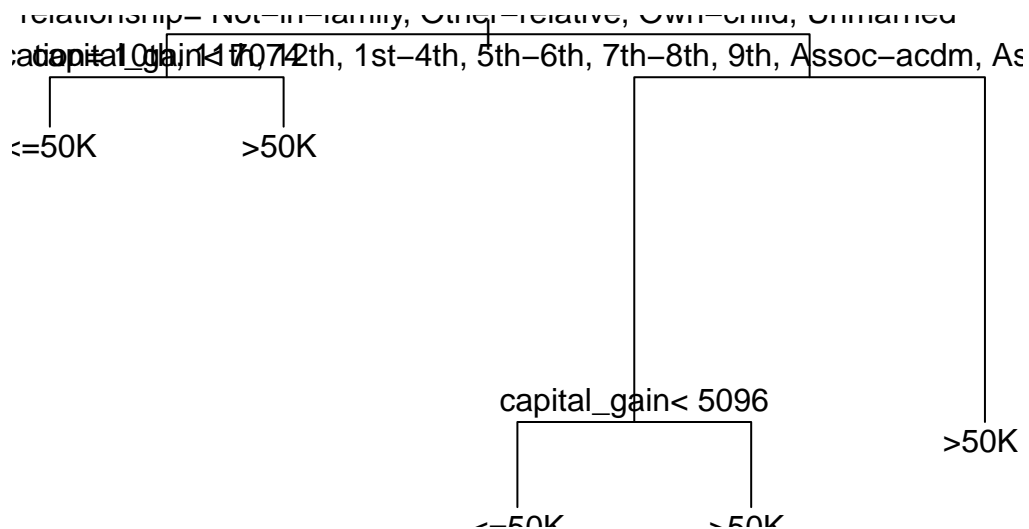
Looking at boxplots and histograms, we also saw that “capital_gain” and “capital_loss” contained many outliers and were heavily skewed. This amount of variability just from the fact that these two variables had most occurrences around 0 but also very large outlier values would severely bias their importance. Although decision trees and random forest are quite robust, it should be a point of attention.

The last variable of interest was “fnlwgt”, which is a statistical metric for each individual. We felt that this was not relevant to the individual’s income level based on the variable’s meaning so we decided to not use it in our analysis.

Looking at boxplots and barplots of various variables, it seems that there is a noticeable difference in age and education of the two income groups. The occupation boxplot showed large differences in income distributions of the respective occupations. These can be seen in the provided images.

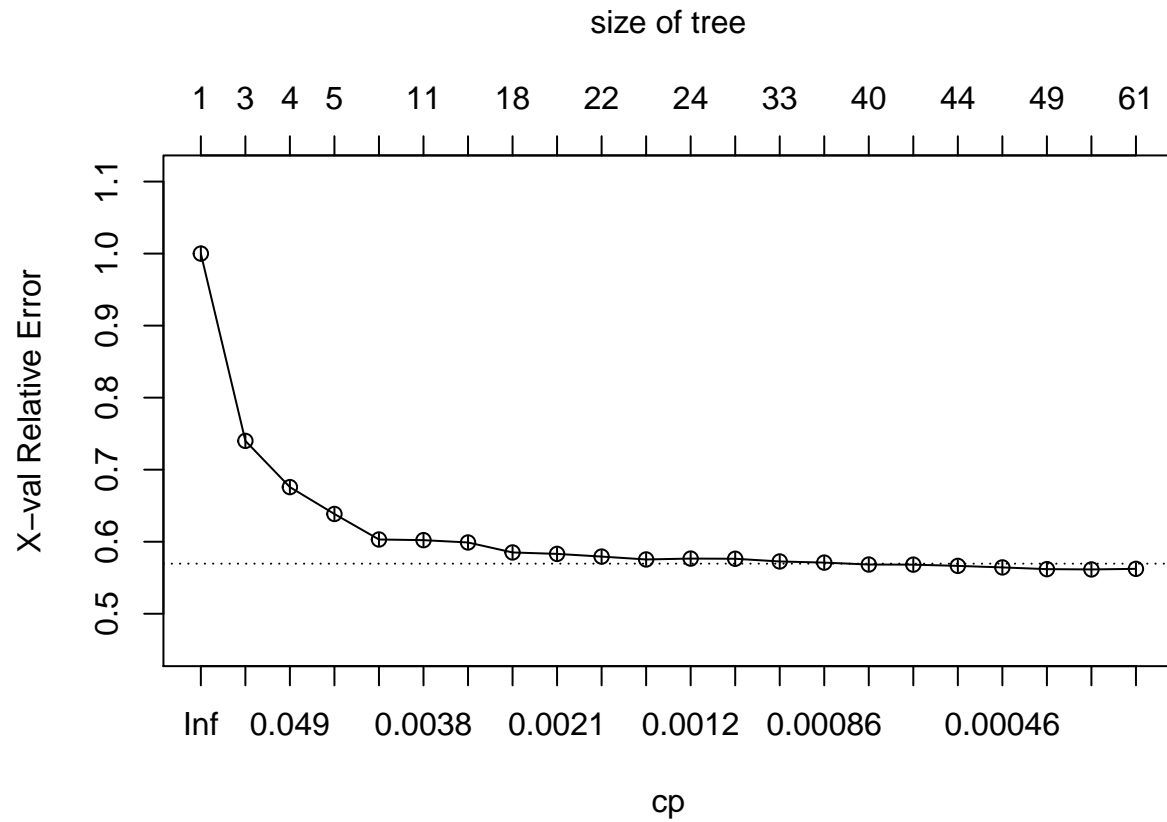
Analysis

Classification Tree:

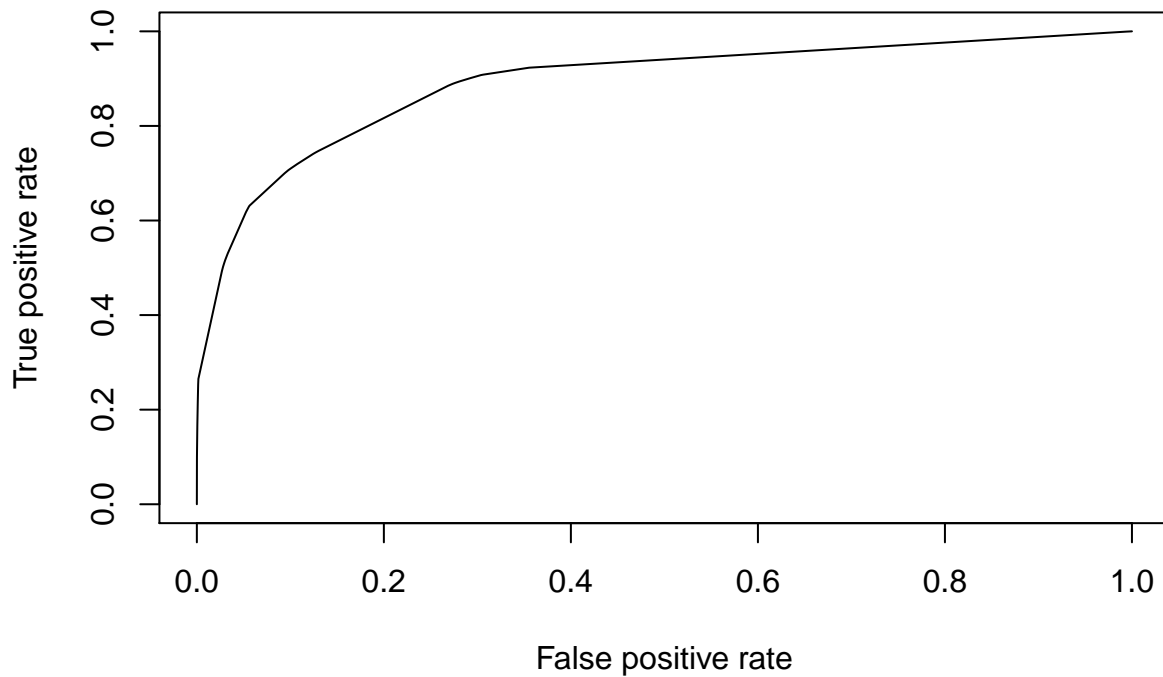


We decided to use the “education_num” variable over the “education” variable as it was cleaner for the tree model. For the tuning, 10-fold cross-validation was used to tune the complexity parameter and the minimum split size, and max depth by the tune.wrapper function in the e1071 package. The error estimation using 10-fold cross validation is 0.1416353.

```
##
## Error estimation of 'rpart.wrapper' using 10-fold cross validation: 0.1416353
```

We found the max depth = 10 , minimum split size =50, cp= 0.0003 minimized the error. We pruned the tree at cp =0.0003. Decreasing CP and increasing minsplit would help to prevent overfitting by growing too large of a tree as CP decreases.



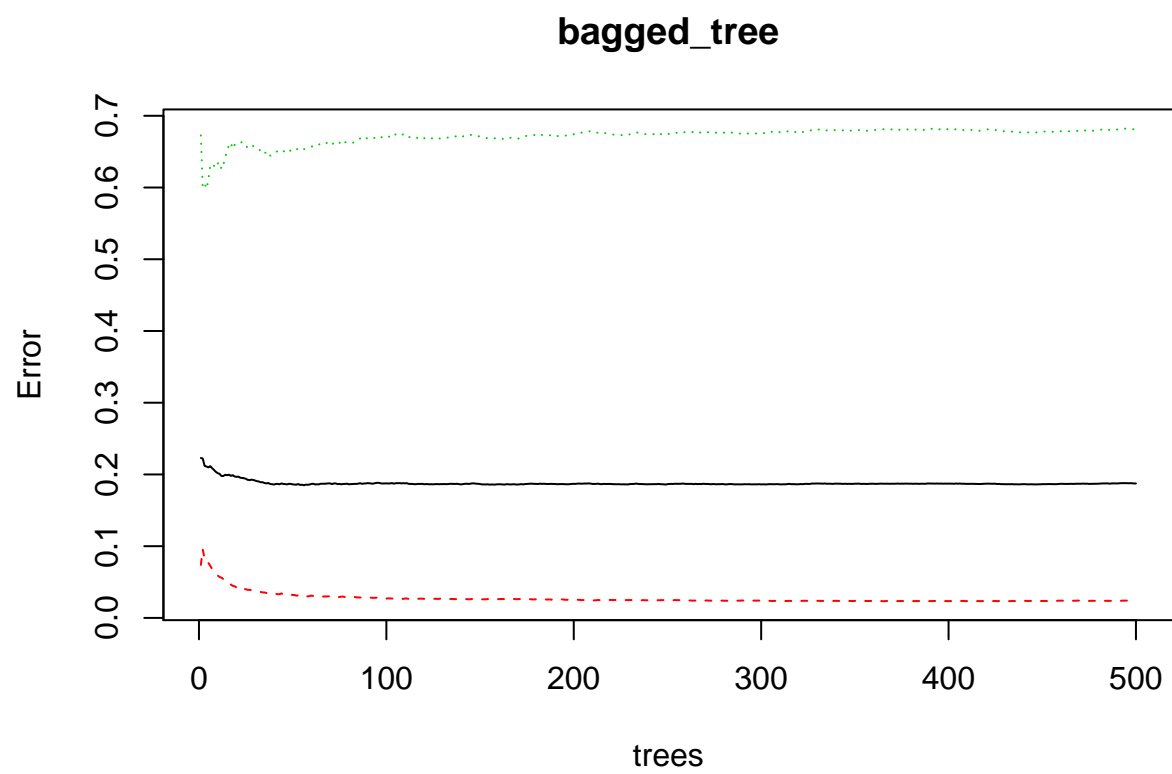
We fit the training data with the combination with the lowest misclassification error, using rpart from the rpart library, then pruned the tree with the CP with the lowest error. This produced a training accuracy of 0.8657. The AUC value was 0.89.

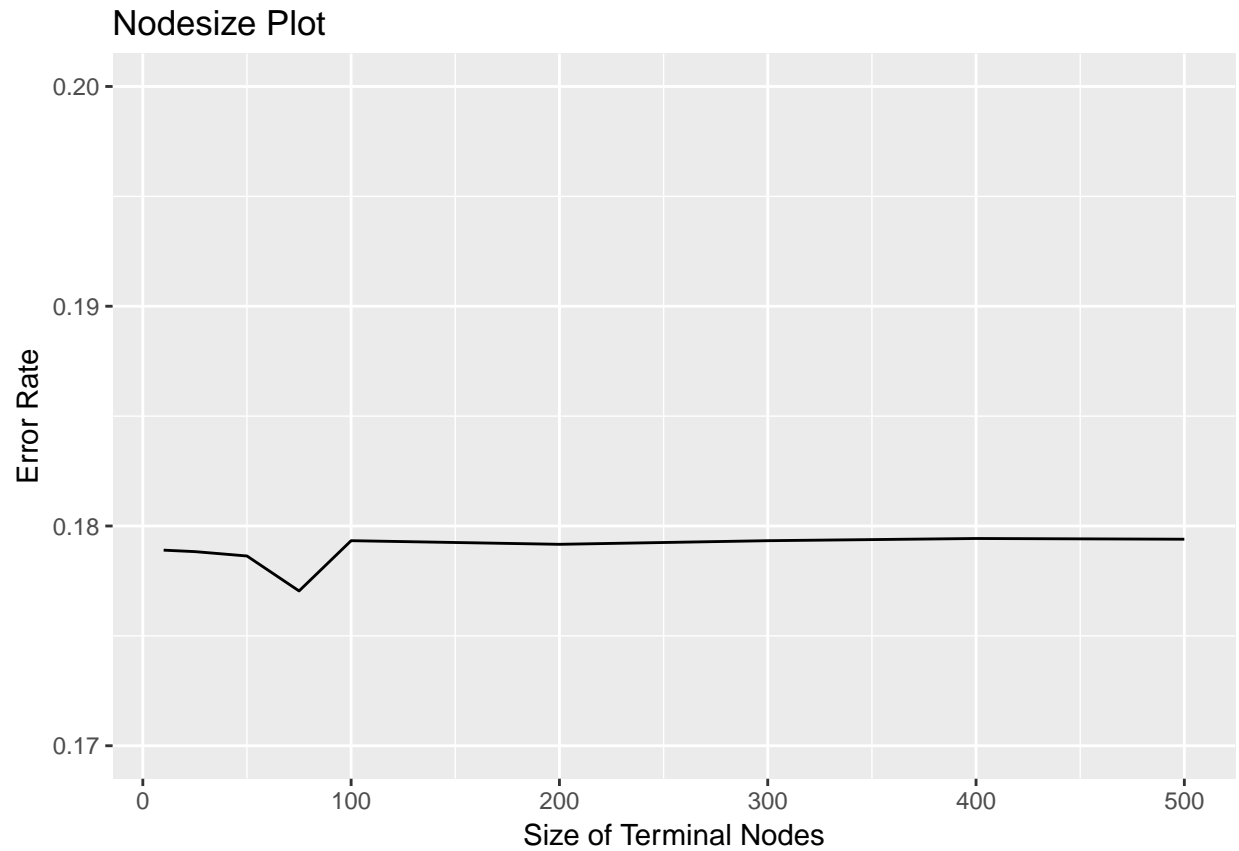
The following are the 7 most important variables as measured by the average decrease in gini:

##	relationship	marital_status	education	capital_gain	occupation
##	2313.3385	2252.7651	1127.6795	1096.4274	957.7924
##	sex	age			
##	770.4282	679.4851			

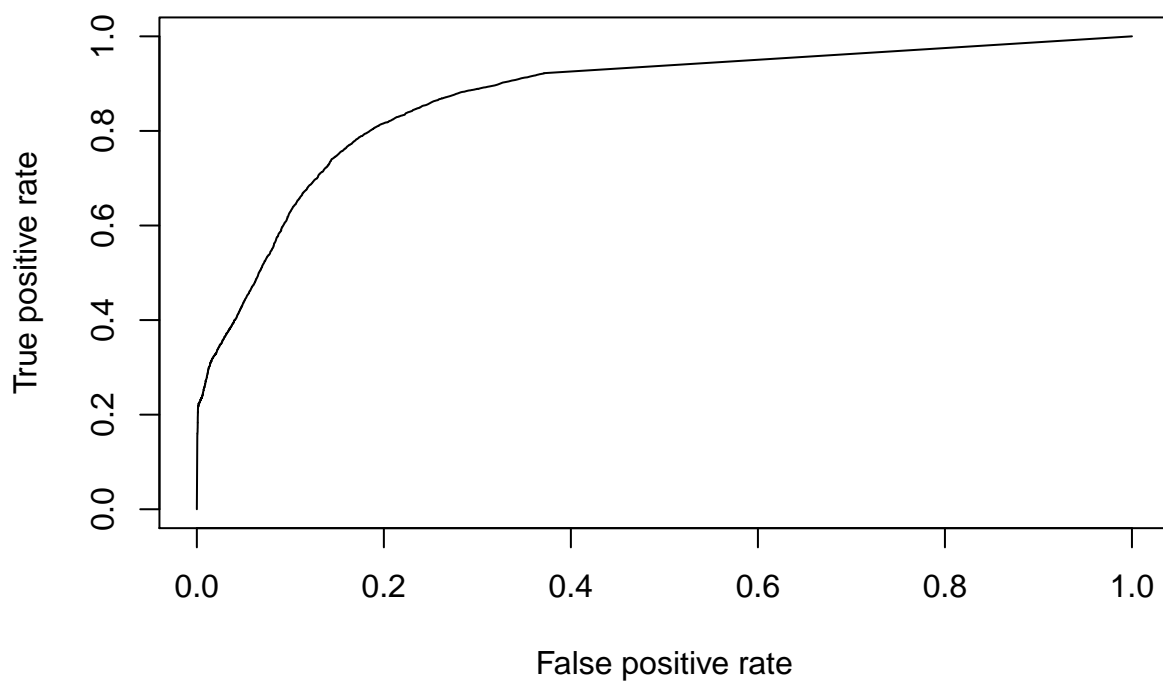
Bagged Tree:

A bagged tree is a special case of a random forest where all the predictors are considered at each split, which is reflected in the mtry parameter. We decided to tune the number of trees and the minimum node size.





It can be seen from the graph of errors and the number of trees that 300 trees stabilizes the error rate. Calculating the errors for different minimum node sizes, the plot shows that around a minimum of 75 produces the optimal error. Choosing 75 allows for controlling the size of the trees.

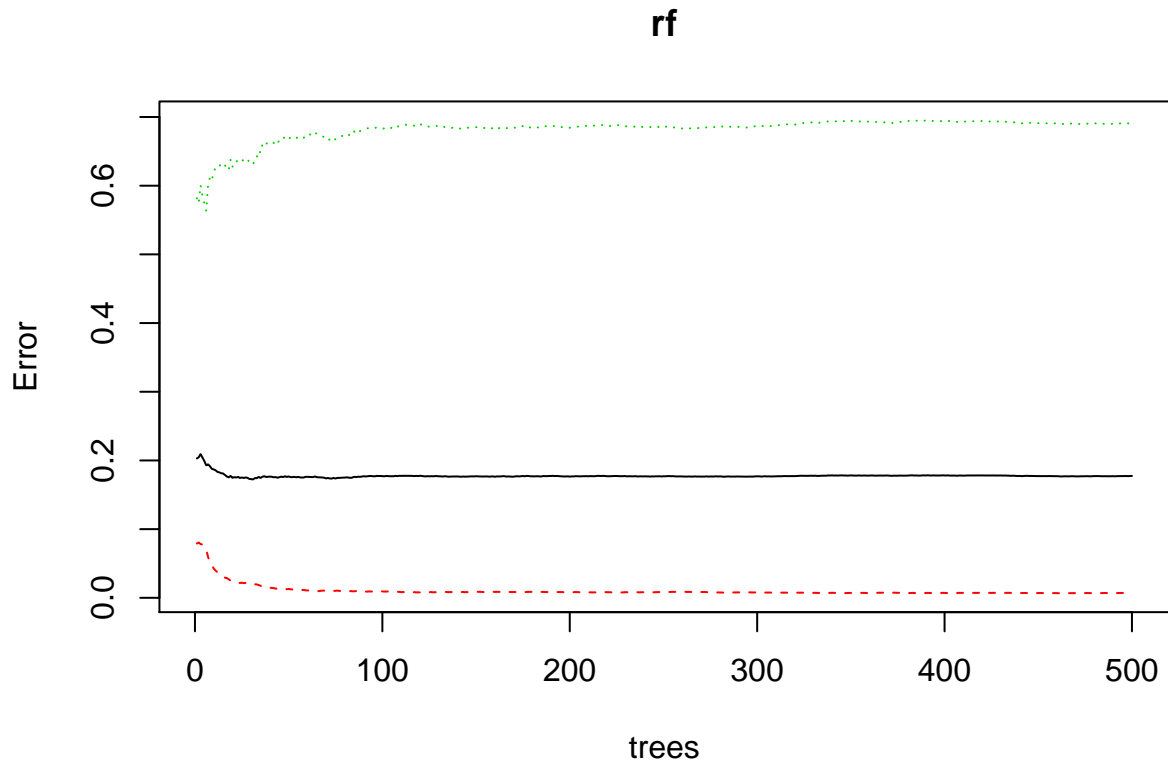


Using randomForest from the randomForest library, we fit the training data, getting a training accuracy of 0.81. The AUC was 0.87.

The seven most important variables were:

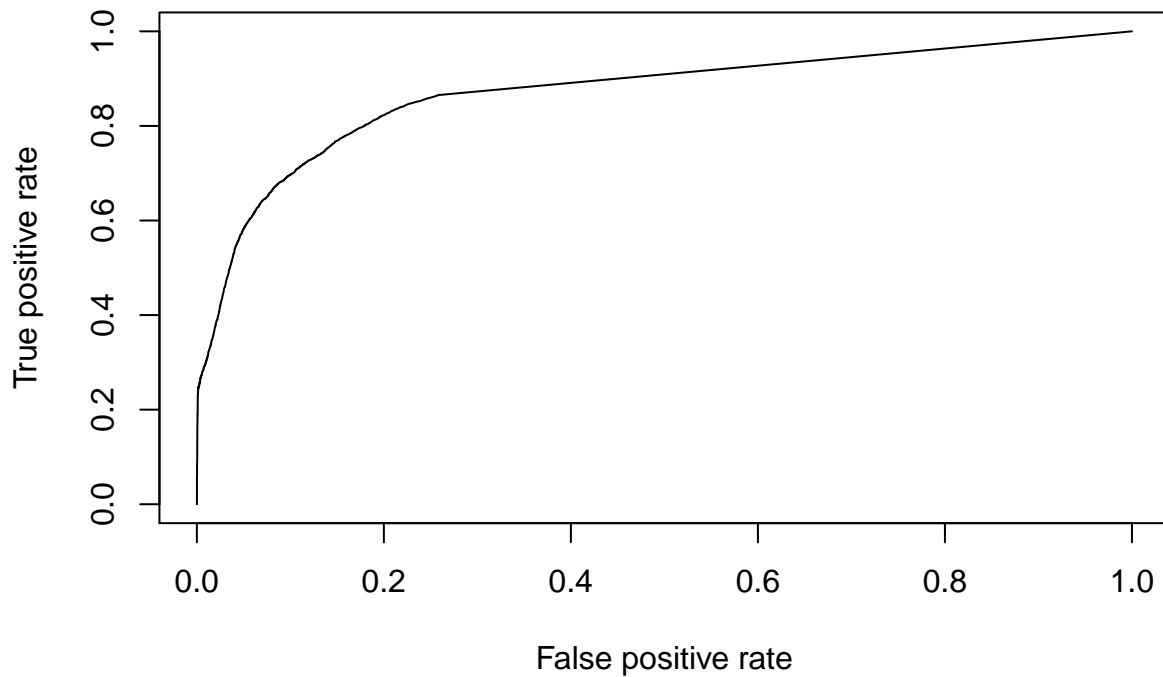
##	relationship	education	capital_gain	occupation	age
##	2294.4472	1143.5576	1102.4702	484.5694	407.3874
##	capital_loss	hours_per_week			
##	308.2347	257.7376			

Random Forest:



A random forest of size 300 trees was fitted using randomForest from the randomForest library. When the number of trees reached 300, three lines are stable and no significant change after 300. 300 trees is sufficient to provide our results with robustness.

Three parameters that are commonly used in tuning are the number of trees, minimum node size (implicitly set the depth of tree) and the number of predictors. We use the random search in MLR package to find the hyperparameter. The hyperparameters with lowest error are mtry:4, nodesize:400, ntree:300. The minimum leaf size was chosen because of the data is fairly unbalanced.

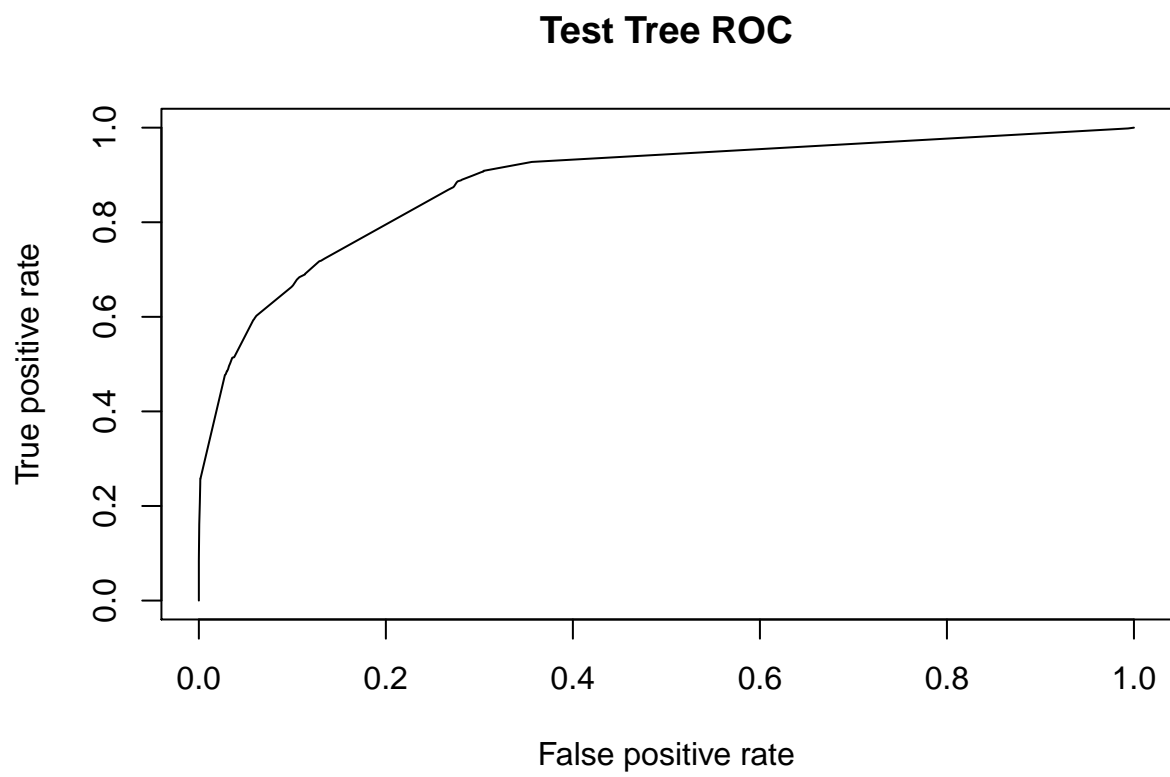


The random forest model has a training accuracy of 0.84, The AUC value was 0.87, lower than that of the tree. It would be useful to look at the false positive rate, which was around 2%.

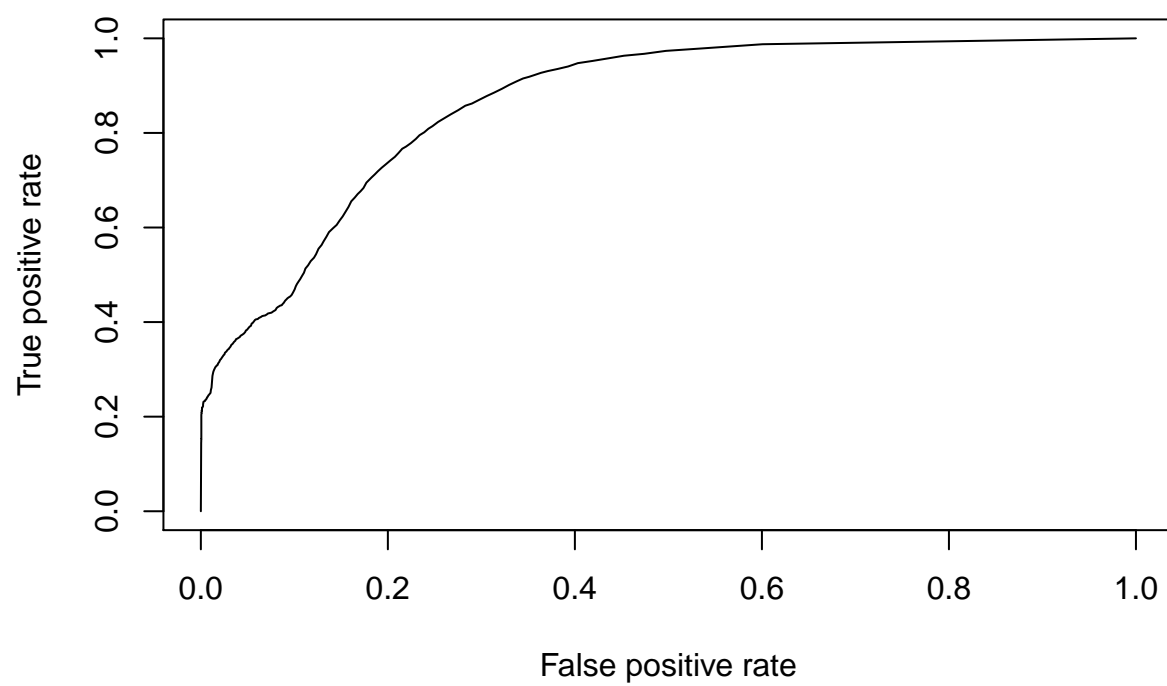
```
## relationship capital_gain marital_status education occupation
## 1234.6693 1006.6616 839.2736 797.1107 587.3411
## age capital_loss
## 281.5575 193.4349
```

Mostly the same variables had the most importance as measured by the mean decrease in gini as for the classification tree, although the values of the mean decrease in gini are noticeably lower. This could be due to the fact that the mtry parameter restricted the selection of variables for splitting.

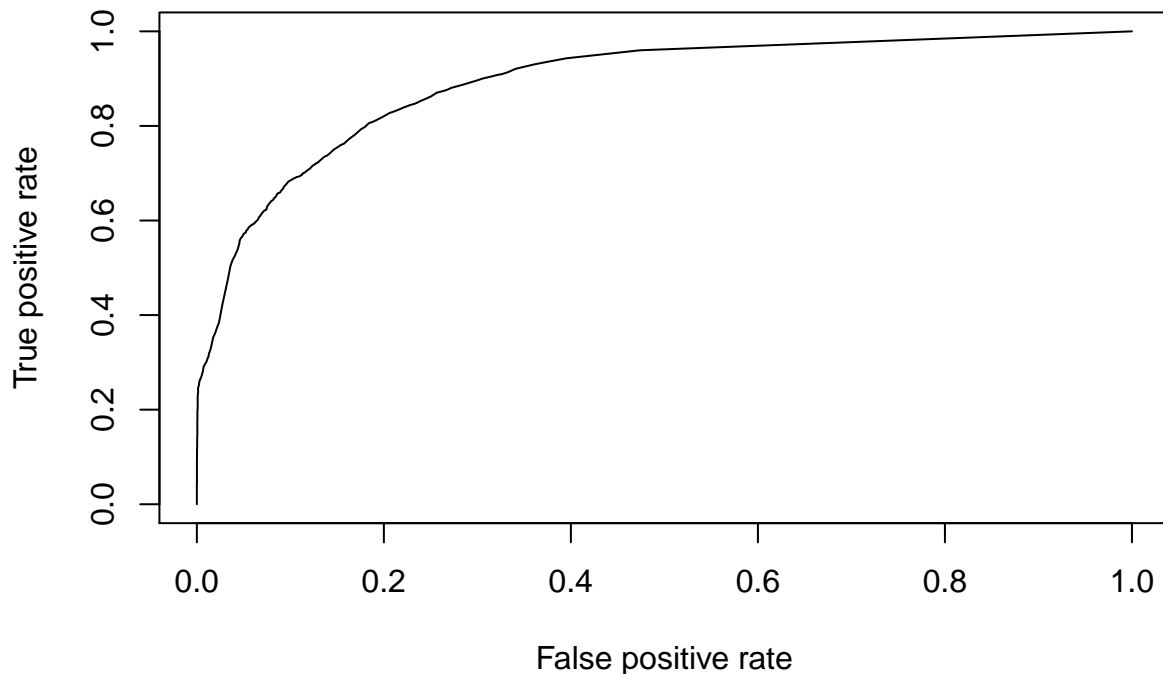
Test Data



Test Bagged Tree ROC



Test Random Forest ROC



Based on the training accuracy the tree is the strongest model. After fitting the tree, the test accuracy was 0.8562 and the AUC for the ROC curve was 0.868. Based on the plots, this is the highest AUC of the three, which would indicate that this classifier addresses the unbalanced data classes the best. The confusion matrix was (observed is the columns, predicted is the rows):

```
tree_confusion_Matrix$table
```

```
##           Reference
## Prediction <=50K >50K
##    <=50K 10679 1485
##    >50K   684 2215
```

The true positive rate (sensitivity) was:

```
## Pos Pred Value
##    0.8779184
```

The true negative rate (specificity) was:

```
## Neg Pred Value
##    0.7640566
```

Conclusion

The classification tree is the strongest classifier of the three methods. The accuracy was 0.856 which was lower than the training accuracy of 0.883. This does indicate some overfitting of the training model. All three models indicated that the most discriminative variables were relationship, education, age, and occupation, although there were some differences in ranking. This is in line with our expectations that age, job, and level

of education are strong indicators of income. It was interesting that race and gender didn't appear to be significant variables for the three models since these are widely thought of as key factors of income inequality.

Due to computational limits, more granular tuning of parameters could not be done, but it would be useful to tune more of the parameters and more values for each parameter to address the unbalanced nature of the data.