Classification

Zachary Canoot & Gray Simpson

Classification

This time we will be using linear models in order to classify observations. Linear models like logistic regression and Naive Bayes work by finding the probability of a target variable given a predictor variable. This means we are predicting a class as opposed to a continuous value like in linear regression. These models are great for data with outliers and are easy to implement and interpret. Linear regression isn't very flexible however, and Naive Bayes makes a naive assumption that predictors are independent.

What is Our Data?

The weather data we used in our quantitative didn't have a suitable categorical target field, so we are switching to income census data. The data has a great binary classification in the form of an IncomeClass attribute that only states whether a given person's income is below or above 50k. We have plenty of categories for each person, and continuous measurements like age and work hours.

The census itself is from the year 1994, and spans various socio-economic groups. We both trying to predict this income classification based on all of the data, as well as just get an understanding of some key predictors in the data.

With IncomeClass as our target, lets analyze the data!

Reading the Data

The data is stored as two files, with rows just delimited by commas, so we read them in to one whole data frame, and label the headers manual using our source as a reference. It's worth noting that this data was extracted with the intention of creating a classification model, so the two files are meant to be training and test data, but we are going to re-distribute the data later.

```
income_train <- read.table("adult.data", sep=",", header=FALSE)
income_test <- read.table("adult.test", sep=",", header=FALSE)
income <- rbind(income_test, income_train)
colnames(income) <- c("Age", "WorkClass", "Weight", "Education", "YearsEdu", "Marital-Status", "Job", "
#Just to check to make sure it read properly
str(income)
```

```
## 'data.frame':    48842 obs. of  15 variables:
##  $ Age           : int  25 38 28 44 18 34 29 63 24 55 ...
##  $ WorkClass     : chr  " Private" " Private" " Local-gov" " Private" ...
##  $ Weight        : int  226802 89814 336951 160323 103497 198693 227026 104626 369667 104996 ...
##  $ Education     : chr  " 11th" " HS-grad" " Assoc-acdm" " Some-college" ...
##  $ YearsEdu      : int  7 9 12 10 10 6 9 15 10 4 ...
##  $ Marital-Status: chr  " Never-married" " Married-civ-spouse" " Married-civ-spouse" " Married-civ-sp
```

```
## $ Job           : chr  " Machine-op-inspct" " Farming-fishing" " Protective-serv" " Machine-op-insp
## $ Relationship  : chr  " Own-child" " Husband" " Husband" " Husband" ...
## $ Race          : chr  " Black" " White" " White" " Black" ...
## $ Sex           : chr  " Male" " Male" " Male" " Male" ...
## $ CapitalGain   : int  0 0 0 7688 0 0 0 3103 0 0 ...
## $ CapitalLoss   : int  0 0 0 0 0 0 0 0 0 0 ...
## $ HoursWorked   : int  40 50 40 40 30 30 40 32 40 10 ...
## $ NativeCountry : chr  " United-States" " United-States" " United-States" " United-States" ...
## $ IncomeClass   : chr  " <=50K." " <=50K." " >50K." " >50K." ...
```

Now we want to turn the qualitative data into factors.

Find all attributes of income that are non-numeric - sapply() returns a logical object of every attribute run through the given function - which() returns all of the true indices of a logical object - income[,] extracts the attributes (See help(Extract)) - We then lapply, with as.factor forcing them to be factors in a list

Then just factor them.

```
# Note here that while sapply returns a vector, lapply returns a list
income[, sapply(income, is.character)] <- lapply(income[, sapply(income, is.character)], as.factor)
# Checking our work
str(income)
```

```
## 'data.frame':    48842 obs. of  15 variables:
## $ Age           : int  25 38 28 44 18 34 29 63 24 55 ...
## $ WorkClass     : Factor w/ 9 levels " ?"," Federal-gov",..: 5 5 3 5 1 5 1 7 5 5 ...
## $ Weight        : int  226802 89814 336951 160323 103497 198693 227026 104626 369667 104996 ...
## $ Education     : Factor w/ 16 levels " 10th"," 11th",..: 2 12 8 16 16 1 12 15 16 6 ...
## $ YearsEdu      : int  7 9 12 10 10 6 9 15 10 4 ...
## $ Marital-Status: Factor w/ 7 levels " Divorced"," Married-AF-spouse",..: 5 3 3 3 5 5 5 3 5 3 ...
## $ Job           : Factor w/ 15 levels " ?"," Adm-clerical",..: 8 6 12 8 1 9 1 11 9 4 ...
## $ Relationship  : Factor w/ 6 levels " Husband"," Not-in-family",..: 4 1 1 1 4 2 5 1 5 1 ...
## $ Race          : Factor w/ 5 levels " Amer-Indian-Eskimo",..: 3 5 5 3 5 5 3 5 5 5 ...
## $ Sex           : Factor w/ 2 levels " Female"," Male": 2 2 2 2 1 2 2 2 1 2 ...
## $ CapitalGain   : int  0 0 0 7688 0 0 0 3103 0 0 ...
## $ CapitalLoss   : int  0 0 0 0 0 0 0 0 0 0 ...
## $ HoursWorked   : int  40 50 40 40 30 30 40 32 40 10 ...
## $ NativeCountry : Factor w/ 42 levels " ?"," Cambodia",..: 40 40 40 40 40 40 40 40 40 40 ...
## $ IncomeClass   : Factor w/ 4 levels " <=50K"," <=50K.",..: 2 2 4 4 2 2 2 4 2 2 ...
```

Now the data is a bit cleaner we can start to look at it!

```
summary(income)
```

```
##       Age                    WorkClass          Weight
##  Min.   :17.00    Private          :33906   Min.   :  12285
##  1st Qu.:28.00    Self-emp-not-inc: 3862   1st Qu.: 117551
##  Median :37.00    Local-gov       : 3136   Median : 178145
##  Mean   :38.64    ?               : 2799   Mean   : 189664
##  3rd Qu.:48.00    State-gov       : 1981   3rd Qu.: 237642
##  Max.   :90.00    Self-emp-inc    : 1695   Max.   :1490400
##                  (Other)          : 1463
##          Education        YearsEdu                    Marital-Status
```

```
##   HS-grad     :15784   Min.   : 1.00    Divorced             : 6633
##   Some-college:10878   1st Qu.: 9.00    Married-AF-spouse    :   37
##   Bachelors   : 8025   Median :10.00    Married-civ-spouse   :22379
##   Masters     : 2657   Mean   :10.08    Married-spouse-absent:  628
##   Assoc-voc   : 2061   3rd Qu.:12.00    Never-married        :16117
##   11th        : 1812   Max.   :16.00    Separated            : 1530
##  (Other)      : 7625                    Widowed              : 1518
##               Job                Relationship                   Race
##   Prof-specialty : 6172    Husband       :19716    Amer-Indian-Eskimo:  470
##   Craft-repair   : 6112    Not-in-family :12583    Asian-Pac-Islander: 1519
##   Exec-managerial: 6086    Other-relative: 1506    Black             : 4685
##   Adm-clerical   : 5611    Own-child     : 7581    Other             :  406
##   Sales          : 5504    Unmarried     : 5125    White             :41762
##   Other-service  : 4923    Wife          : 2331
##  (Other)         :14434
##      Sex         CapitalGain      CapitalLoss      HoursWorked
##   Female:16192   Min.   :    0   Min.   :   0.0   Min.   : 1.00
##   Male  :32650   1st Qu.:    0   1st Qu.:   0.0   1st Qu.:40.00
##                  Median :    0   Median :   0.0   Median :40.00
##                  Mean   : 1079   Mean   :  87.5   Mean   :40.42
##                  3rd Qu.:    0   3rd Qu.:   0.0   3rd Qu.:45.00
##                  Max.   :99999   Max.   :4356.0   Max.   :99.00
##
##        NativeCountry    IncomeClass
##   United-States:43832   <=50K :24720
##   Mexico       :  951   <=50K.:12435
##   ?            :  857   >50K  : 7841
##   Philippines  :  295   >50K. : 3846
##   Germany      :  206
##   Puerto-Rico  :  184
##  (Other)       : 2517
```

Now that we can really see our factor's options, I see a couple skewed data points: - Twice as many men as women! Hope those numbers are better in 2022! - A large percent of the data is for natives to the US, which is kind of expected - Weight: Now, this represent what census takers thought a particular row represented the whole of the dataset. I must admit at the time I don't know how to account for statistical weight, but considering our model only needs to match training data, not other data from 1994, we are safe to ignore it.

The data looks very clean! Except for a bit of an anomaly with how the Target column, IncomeClass is stored. Some levels have a "." at the end, which we would like to remove. So lets go ahead and condense that, remove the Weight attribute, and create our training and test data.

```
# Simply just reassign the levels
levels(income$IncomeClass) <- c("<=50k", "<=50k", ">50k", ">50k")
levels(income$IncomeClass)
```

```
## [1] "<=50k" ">50k"
```

```
# Then remove the attribute weight using it's index
income <- income[, -3]
str(income)
```

```
## 'data.frame':    48842 obs. of  14 variables:
```

```
##  $ Age           : int  25 38 28 44 18 34 29 63 24 55 ...
##  $ WorkClass     : Factor w/ 9 levels " ?"," Federal-gov",..: 5 5 3 5 1 5 1 7 5 5 ...
##  $ Education     : Factor w/ 16 levels " 10th"," 11th",..: 2 12 8 16 16 1 12 15 16 6 ...
##  $ YearsEdu      : int  7 9 12 10 10 6 9 15 10 4 ...
##  $ Marital-Status: Factor w/ 7 levels " Divorced"," Married-AF-spouse",..: 5 3 3 3 5 5 5 3 5 3 ...
##  $ Job           : Factor w/ 15 levels " ?"," Adm-clerical",..: 8 6 12 8 1 9 1 11 9 4 ...
##  $ Relationship  : Factor w/ 6 levels " Husband"," Not-in-family",..: 4 1 1 1 4 2 5 1 5 1 ...
##  $ Race          : Factor w/ 5 levels " Amer-Indian-Eskimo",..: 3 5 5 3 5 5 3 5 5 5 ...
##  $ Sex           : Factor w/ 2 levels " Female"," Male": 2 2 2 2 1 2 2 2 1 2 ...
##  $ CapitalGain   : int  0 0 0 7688 0 0 0 3103 0 0 ...
##  $ CapitalLoss   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ HoursWorked   : int  40 50 40 40 30 30 40 32 40 10 ...
##  $ NativeCountry : Factor w/ 42 levels " ?"," Cambodia",..: 40 40 40 40 40 40 40 40 40 40 ...
##  $ IncomeClass   : Factor w/ 2 levels "<=50k",">50k": 1 1 2 2 1 1 1 2 1 1 ...
```

Then we are good to start exploring!

Training Data Exploration

Spliting Training Data

We are splitting training data on a 80/20 split

```
set.seed(42069)
trainindex <- sample(1:nrow(income),nrow(income)*.8,replace=FALSE)
train <- income[trainindex,]
test <- income[-trainindex,]
# Cleaning up earlier data
rm("income", "income_test", "income_train")
```

Textual Measurements

And what does that training data look like!

We would want to use different metrics, like mean, or count our factors:

```
mean(train$Age)
```

```
## [1] 38.63335
```

```
nlevels(train$WorkClass)
```

```
## [1] 9
```

But we can just do that in `summary()`.

```
summary(train)
```

```
##       Age                     WorkClass            Education    
##  Min.   :17.00    Private         :27152    HS-grad     :12633  
##  1st Qu.:28.00    Self-emp-not-inc: 3081    Some-college: 8704  
##  Median :37.00    Local-gov       : 2550    Bachelors   : 6385  
##  Mean   :38.63    ?               : 2204    Masters     : 2178  
##  3rd Qu.:48.00    State-gov       : 1551    Assoc-voc   : 1642  
##  Max.   :90.00    Self-emp-inc    : 1348    11th        : 1447  
##                  (Other)          : 1187   (Other)      : 6084  
##     YearsEdu                    Marital-Status              Job       
##  Min.   : 1.00    Divorced             : 5307    Prof-specialty : 4977  
##  1st Qu.: 9.00    Married-AF-spouse    :   33    Craft-repair   : 4908  
##  Median :10.00    Married-civ-spouse   :17860    Exec-managerial: 4800  
##  Mean   :10.08    Married-spouse-absent:  507    Adm-clerical   : 4522  
##  3rd Qu.:12.00    Never-married        :12883    Sales          : 4386  
##  Max.   :16.00    Separated            : 1243    Other-service  : 3964  
##                   Widowed              : 1240   (Other)         :11516  
##           Relationship                    Race             Sex       
##   Husband       :15726    Amer-Indian-Eskimo:  379    Female:13039  
##   Not-in-family :10070    Asian-Pac-Islander: 1189    Male  :26034  
##   Other-relative: 1186    Black             : 3737                  
##   Own-child     : 6072    Other             :  322                  
##   Unmarried     : 4140    White             :33446                  
##   Wife          : 1879                                              
##                                                                     
##   CapitalGain     CapitalLoss       HoursWorked         NativeCountry  
##  Min.   :    0   Min.   :   0.00   Min.   : 1.00    United-States:35070  
##  1st Qu.:    0   1st Qu.:   0.00   1st Qu.:40.00    Mexico       :  766  
##  Median :    0   Median :   0.00   Median :40.00    ?            :  688  
##  Mean   : 1055   Mean   :  87.39   Mean   :40.42    Philippines  :  228  
##  3rd Qu.:    0   3rd Qu.:   0.00   3rd Qu.:45.00    Germany      :  164  
##  Max.   :99999   Max.   :4356.00   Max.   :99.00    Canada       :  144  
##                                                    (Other)       : 2013  
##  IncomeClass 
##  <=50k:29759  
##  >50k : 9314  
##                
##                
##                
##                
##                
```

The summary above is good for making sure there is no errors in the data, and of course skews we can deal with. For this data, there sure are a lot of men native to America, but that as said earlier is expected. Looking a bit more:

```
sum(is.na(train))
```

```
## [1] 0
```

```
head(train)
```

```
##       Age    WorkClass    Education YearsEdu      Marital-Status
## 8990   21      Private      HS-grad        9       Never-married
```

```
## 37354  56  Federal-gov    Bachelors       13       Never-married
## 36204  33      Private      HS-grad        9  Married-civ-spouse
## 116    26      Private      HS-grad        9       Never-married
## 6500   30            ?    Assoc-voc       11       Never-married
## 34793  33      Private  Prof-school       15  Married-civ-spouse
##                     Job   Relationship   Race     Sex CapitalGain CapitalLoss
## 8990    Transport-moving      Own-child  White    Male           0           0
## 37354   Transport-moving  Not-in-family  Black    Male           0        2001
## 36204   Transport-moving        Husband  White    Male        3908           0
## 116    Handlers-cleaners      Unmarried  White    Male           0           0
## 6500                   ?  Not-in-family  White    Male           0           0
## 34793    Exec-managerial           Wife  White  Female           0           0
##       HoursWorked  NativeCountry IncomeClass
## 8990           40  United-States       <=50k
## 37354          65  United-States       <=50k
## 36204          40  United-States       <=50k
## 116            40  United-States       <=50k
## 6500           30  United-States       <=50k
## 34793          40  United-States        >50k
```

```
tail(train)
```

```
##       Age         WorkClass     Education YearsEdu      Marital-Status
## 17194  53  Self-emp-not-inc          10th        6  Married-civ-spouse
## 28300  45           Private     Doctorate       16  Married-civ-spouse
## 7694   21           Private       HS-grad        9       Never-married
## 7748   34           Private       HS-grad        9  Married-civ-spouse
## 23193  65  Self-emp-not-inc  Some-college       10  Married-civ-spouse
## 27792  27         State-gov     Bachelors       13       Never-married
##                     Job    Relationship   Race   Sex CapitalGain CapitalLoss
## 17194    Farming-fishing         Husband  White  Male           0           0
## 28300     Prof-specialty         Husband  White  Male        7688           0
## 7694        Adm-clerical  Other-relative  White  Male           0           0
## 7748       Other-service         Husband  White  Male           0           0
## 23193  Machine-op-inspct         Husband  White  Male        6514           0
## 27792     Prof-specialty   Not-in-family  White  Male           0           0
##       HoursWorked  NativeCountry IncomeClass
## 17194          60  United-States       <=50k
## 28300          50  United-States        >50k
## 7694           40  United-States       <=50k
## 7748           40  United-States        >50k
## 23193          40  United-States        >50k
## 27792          30  United-States       <=50k
```

We get an example of whats at the end and start of the data set, and make sure there are no NA's. The census people really keep their data clean.

For one more look lets see some correlation data. Curious how much capital loss went up with age? We can see below… well not much honestly.

```
cor(train$Age, train$CapitalLoss)
```

```
## [1] 0.05806488
```

Text Analysis Conclusion We fear the skew of my data towards 1 type of person (Married Men about to hit their 40's) will make the model's we produce perform well for our dataset, but fail to get any real world accuracy. Obviously if this model was actually destined to predict in the real world if people's income was above or below a certain level (in the 1990's), well if we had all this data we would probably already know their income. So the model is a pointless but fun experiment…

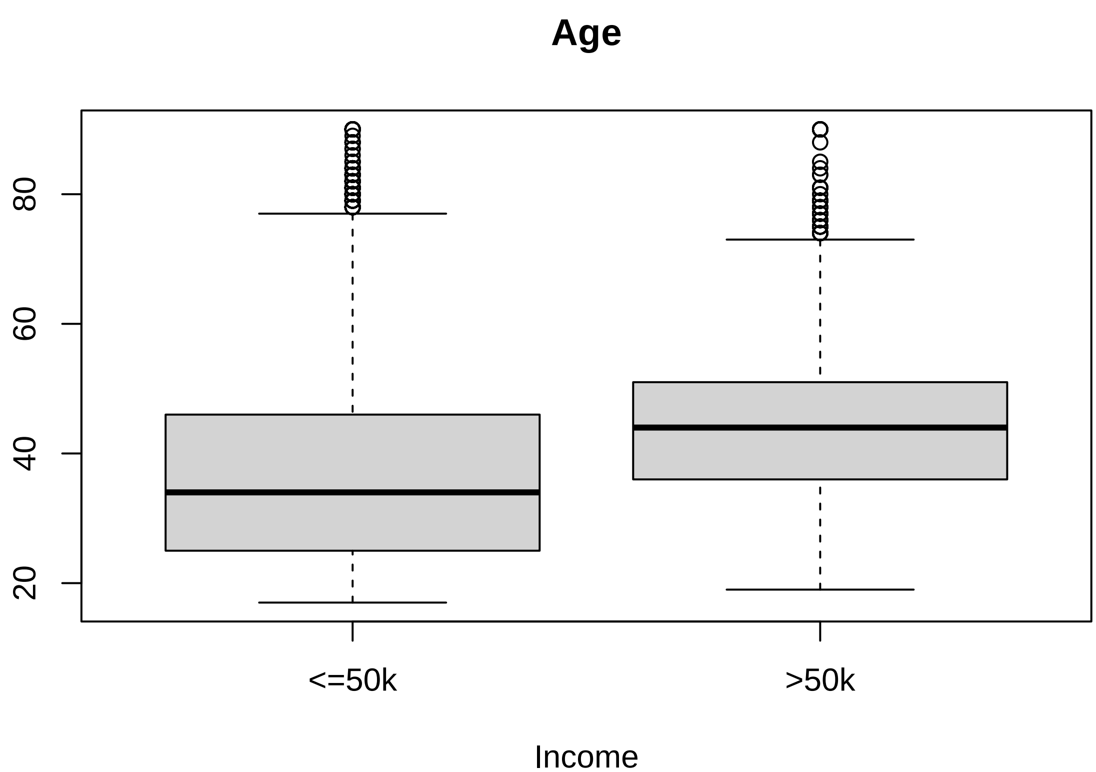
Regardless it is worth noting that a transformation of the data before running logistic regression or naive bayes could produce better results, but it is beyond the scope of this experiment.

While it is probably a realistic distribution of income class (3 people with less than 50k for every person over 50k), the data may just guess that everyone doesn't make that much money due to the skew. This actually is a lot more important then skewed predictors, as our eventual precision/recall could be quite bad. For now, simply observing this is good enough, but this should be onsidered for the final analysis. (And perhaps in our comparision between Bayes and logistic regression).

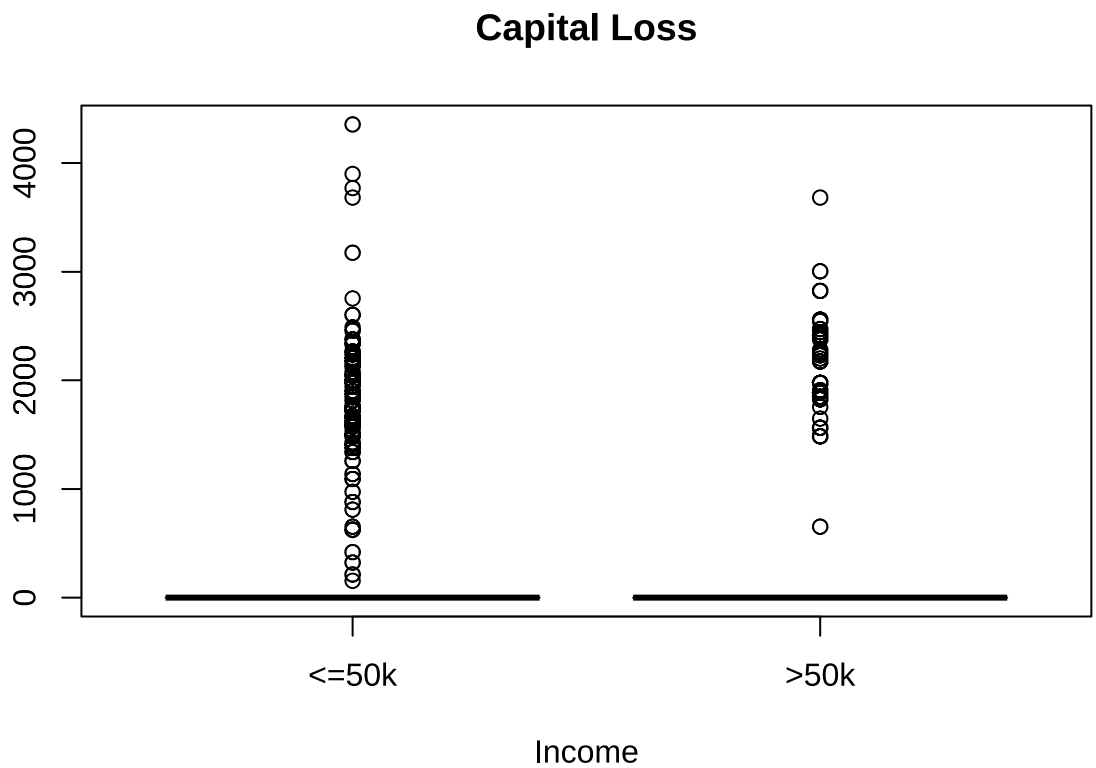
Visual Analysis

We want to see how our target, IncomeClass relates to our numerical data:

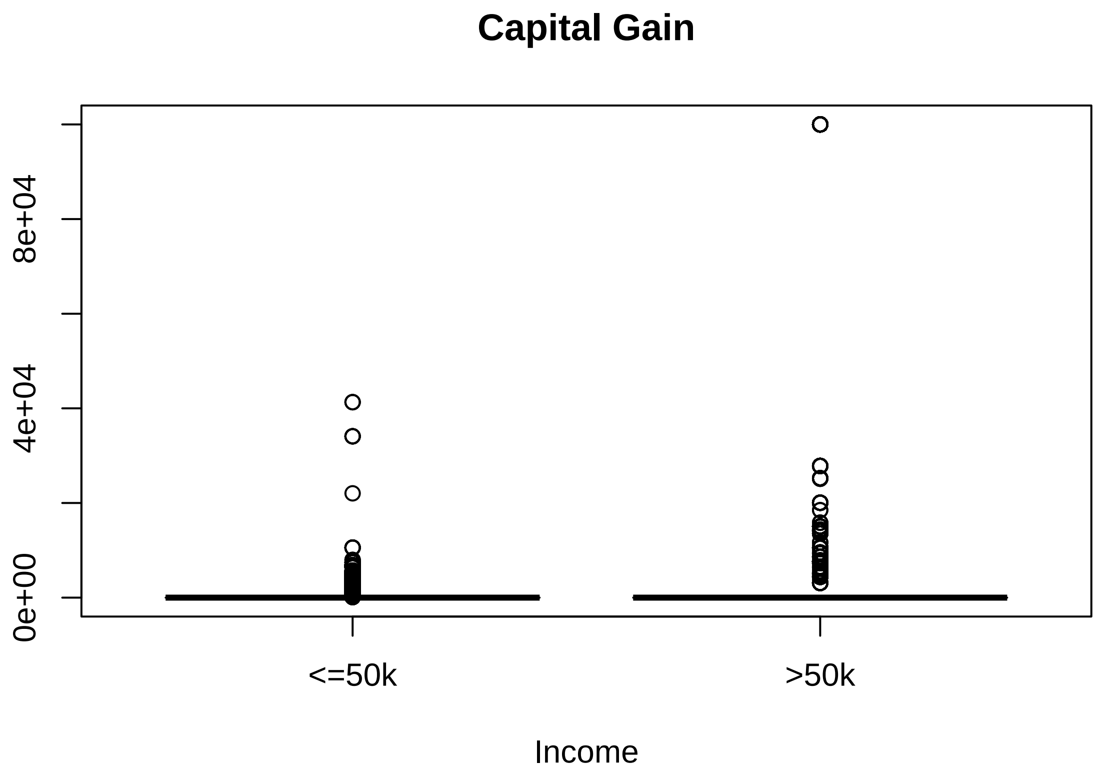
```
plot(x = train$IncomeClass, y=train$Age, ylab="", xlab="Income", main="Age")
```



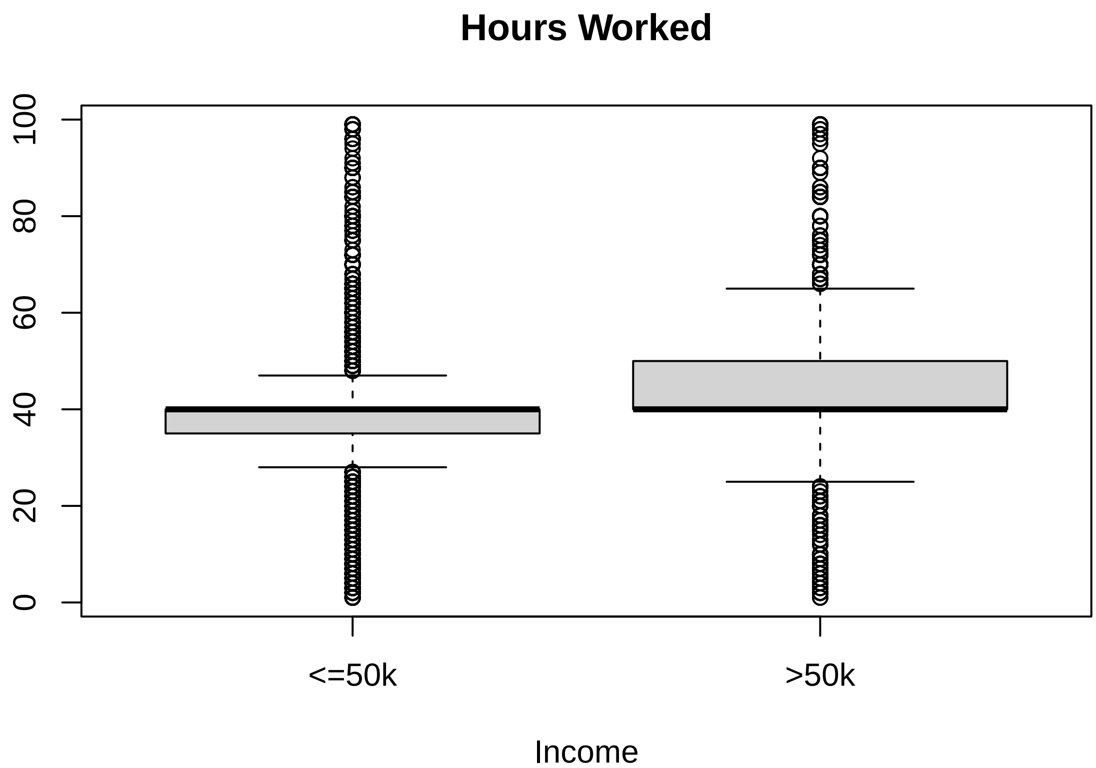

```
plot(x = train$IncomeClass, y=train$CapitalLoss, ylab="", xlab="Income", main="Capital Loss")
```




```r
plot(x = train$IncomeClass, y=train$CapitalGain, ylab="", xlab="Income", main="Capital Gain")
```



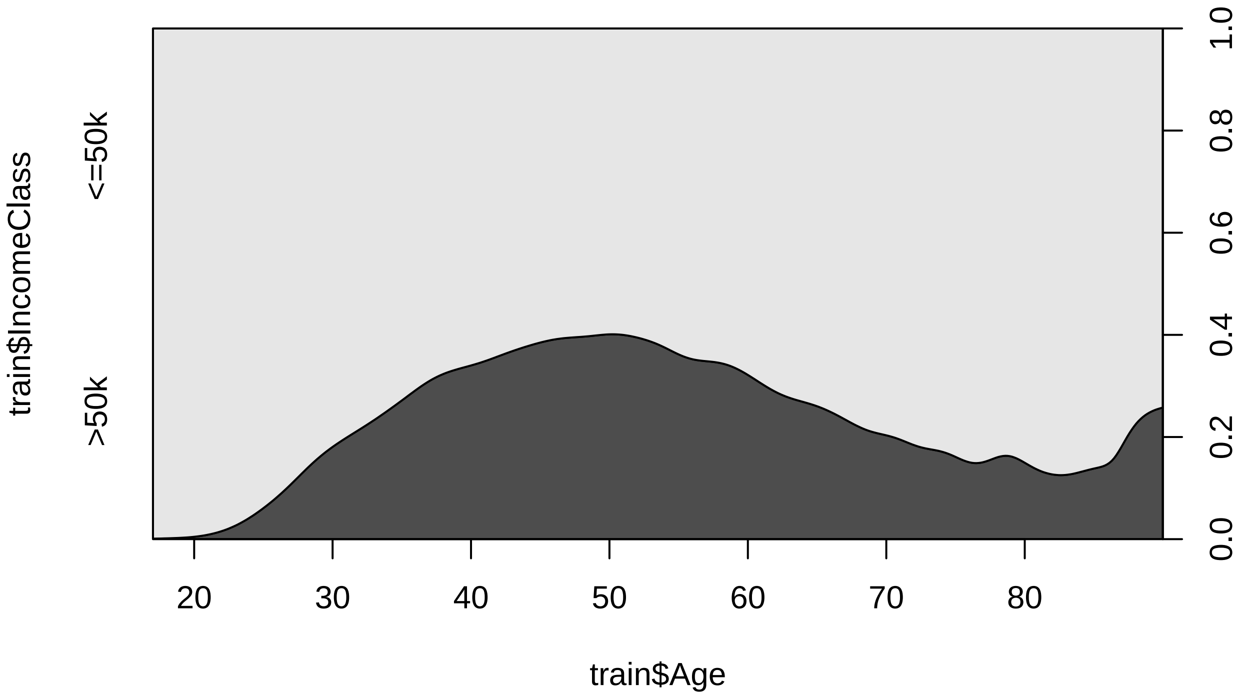

```
plot(x = train$IncomeClass, y=train$HoursWorked, ylab="", xlab="Income", main="Hours Worked")
```



Numerical trends are just easier to spot, especially the effect of age on IncomeClass. You can definitely see in the ease graphs, particular age and hours worked, that there are *some* grounds to predict this income classification based on the predictor data.

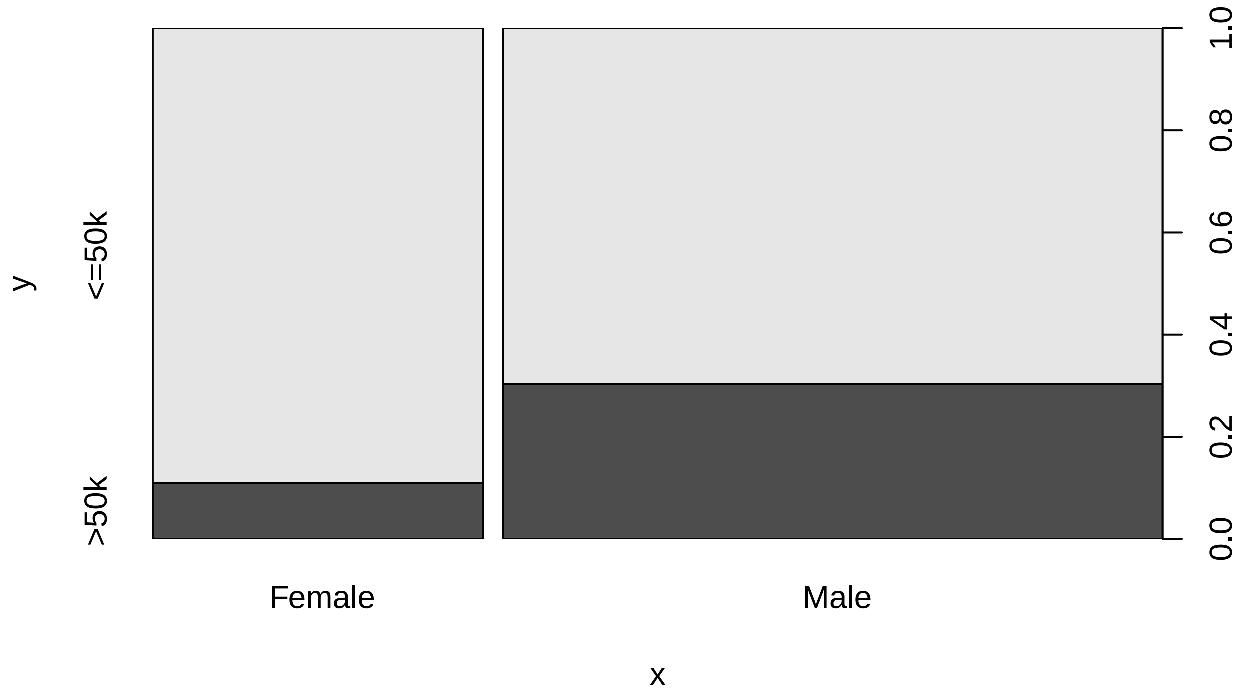
For another view:

```
cdplot(train$Age, train$IncomeClass)
```




```
breaks <- (0:10)*10
plot(train$IncomeClass ~ findInterval(train$HoursWorked, breaks))
```


```
plot(train$Sex, train$IncomeClass)
```



Above we can see a couple trends relating to Income Class: - Women don't make as much as men - It seems the more hours worked, the higher your chances of making it over 50k - Right around 50 years old is when people were the most likely to make >50k

Visual Analysis Conclusion There are so many different factors in this data, that we think assuming the factors are independent could harm the eventual accuracy of our linear models. While we can graph individual factors relation to the target, there are complicated relationships between the predictor data. We may be able to guess that more education would lead to a higher income, but an in-depth analysis of how gender or native country may hamper access to education isnt represented by just the relationship from gender to income. To the final product, it just *looks* like you can bet women make less money, even if that may be due to a compaction of other factors.

Just a couple trends are seen above, and they still tell us that there is some merit to this data being alble to predict relations between our predictors and our target. Now it is time to see if all of those predictors together have a good chance of classifying them into the >50k or <=50k levels.

Classification Regression

Logistic Regression

```
glm1 <- glm(IncomeClass~., data=train, family=binomial)
```

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

```
summary(glm1)
```

```
##
## Call:
## glm(formula = IncomeClass ~ ., family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.0090  -0.4995  -0.1830  -0.0345   3.7816
##
## Coefficients: (2 not defined because of singularities)
##                                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)                          -8.615e+00  4.031e-01 -21.375  < 2e-16
## Age                                   2.582e-02  1.506e-03  17.137  < 2e-16
## WorkClass Federal-gov                 1.117e+00  1.414e-01   7.899 2.81e-15
## WorkClass Local-gov                   4.959e-01  1.292e-01   3.839 0.000123
## WorkClass Never-worked               -9.212e+00  1.469e+02  -0.063 0.950002
## WorkClass Private                     6.406e-01  1.151e-01   5.566 2.61e-08
## WorkClass Self-emp-inc                7.979e-01  1.374e-01   5.808 6.32e-09
## WorkClass Self-emp-not-inc            1.010e-01  1.258e-01   0.803 0.422234
## WorkClass State-gov                   2.780e-01  1.393e-01   1.996 0.045932
## WorkClass Without-pay                -1.159e-01  8.085e-01  -0.143 0.886064
## Education 11th                       -7.159e-03  1.916e-01  -0.037 0.970197
## Education 12th                        3.863e-01  2.353e-01   1.642 0.100616
## Education 1st-4th                    -8.048e-01  4.807e-01  -1.674 0.094109
## Education 5th-6th                    -1.813e-01  2.833e-01  -0.640 0.522149
## Education 7th-8th                    -5.719e-01  2.109e-01  -2.712 0.006697
## Education 9th                        -3.195e-01  2.382e-01  -1.342 0.179742
## Education Assoc-acdm                  1.361e+00  1.596e-01   8.529  < 2e-16
## Education Assoc-voc                   1.230e+00  1.537e-01   8.007 1.18e-15
## Education Bachelors                   1.876e+00  1.424e-01  13.172  < 2e-16
## Education Doctorate                   2.782e+00  1.947e-01  14.292  < 2e-16
## Education HS-grad                     7.575e-01  1.385e-01   5.468 4.55e-08
## Education Masters                     2.217e+00  1.512e-01  14.659  < 2e-16
## Education Preschool                  -4.759e+00  3.380e+00  -1.408 0.159147
## Education Prof-school                 2.665e+00  1.827e-01  14.590  < 2e-16
## Education Some-college                1.125e+00  1.406e-01   7.998 1.26e-15
## YearsEdu                                     NA         NA      NA       NA
## `Marital-Status` Married-AF-spouse    2.347e+00  5.025e-01   4.671 2.99e-06
## `Marital-Status` Married-civ-spouse   2.409e+00  2.491e-01   9.673  < 2e-16
## `Marital-Status` Married-spouse-absent -2.639e-02  2.117e-01  -0.125 0.900790
## `Marital-Status` Never-married       -3.386e-01  8.077e-02  -4.192 2.76e-05
## `Marital-Status` Separated           -1.459e-02  1.512e-01  -0.096 0.923134
## `Marital-Status` Widowed              1.391e-02  1.421e-01   0.098 0.922018
## Job Adm-clerical                      1.151e-01  9.015e-02   1.277 0.201739
## Job Armed-Forces                      4.918e-01  9.198e-01   0.535 0.592853
## Job Craft-repair                      1.499e-01  7.761e-02   1.932 0.053395
## Job Exec-managerial                   8.636e-01  7.999e-02  10.796  < 2e-16
## Job Farming-fishing                  -9.033e-01  1.306e-01  -6.918 4.59e-12
## Job Handlers-cleaners                -5.781e-01  1.307e-01  -4.425 9.66e-06
## Job Machine-op-inspct                -2.374e-01  9.802e-02  -2.422 0.015422
## Job Other-service                    -7.080e-01  1.128e-01  -6.276 3.48e-10
## Job Priv-house-serv                  -2.576e+00  1.060e+00  -2.430 0.015098
```

```
## Job Prof-specialty                        6.595e-01  8.551e-02   7.713 1.23e-14
## Job Protective-serv                       6.323e-01  1.196e-01   5.285 1.26e-07
## Job Sales                                 3.667e-01  8.268e-02   4.435 9.19e-06
## Job Tech-support                          6.302e-01  1.085e-01   5.810 6.23e-09
## Job Transport-moving                             NA         NA      NA       NA
## Relationship Not-in-family                5.632e-01  2.465e-01   2.285 0.022341
## Relationship Other-relative              -4.990e-01  2.234e-01  -2.234 0.025510
## Relationship Own-child                   -5.416e-01  2.401e-01  -2.255 0.024105
## Relationship Unmarried                    3.539e-01  2.619e-01   1.351 0.176604
## Relationship Wife                         1.119e+00  9.301e-02  12.031  < 2e-16
## Race Asian-Pac-Islander                   6.698e-01  2.425e-01   2.763 0.005732
## Race Black                                3.964e-01  2.072e-01   1.914 0.055678
## Race Other                                4.413e-01  3.037e-01   1.453 0.146207
## Race White                                5.388e-01  1.970e-01   2.735 0.006239
## Sex Male                                  6.887e-01  7.157e-02   9.622  < 2e-16
## CapitalGain                               3.127e-04  9.310e-06  33.588  < 2e-16
## CapitalLoss                               6.570e-04  3.397e-05  19.341  < 2e-16
## HoursWorked                               2.785e-02  1.460e-03  19.077  < 2e-16
## NativeCountry Cambodia                    6.302e-01  5.983e-01   1.053 0.292205
## NativeCountry Canada                      5.840e-01  2.598e-01   2.248 0.024601
## NativeCountry China                      -5.256e-01  3.473e-01  -1.513 0.130198
## NativeCountry Columbia                   -2.340e+00  7.308e-01  -3.202 0.001366
## NativeCountry Cuba                        5.021e-02  3.298e-01   0.152 0.878996
## NativeCountry Dominican-Republic         -1.520e+00  7.539e-01  -2.016 0.043755
## NativeCountry Ecuador                     2.593e-02  6.061e-01   0.043 0.965877
## NativeCountry El-Salvador                -5.357e-01  4.807e-01  -1.114 0.265081
## NativeCountry England                     4.262e-01  3.182e-01   1.339 0.180450
## NativeCountry France                      9.755e-01  5.350e-01   1.823 0.068239
## NativeCountry Germany                    -9.307e-02  2.737e-01  -0.340 0.733794
## NativeCountry Greece                     -5.508e-01  4.557e-01  -1.209 0.226708
## NativeCountry Guatemala                  -3.423e-01  7.380e-01  -0.464 0.642786
## NativeCountry Haiti                      -2.569e-01  5.923e-01  -0.434 0.664456
## NativeCountry Holand-Netherlands         -9.569e+00  5.354e+02  -0.018 0.985741
## NativeCountry Honduras                    1.175e-01  1.136e+00   0.103 0.917586
## NativeCountry Hong                       -8.128e-01  6.816e-01  -1.192 0.233072
## NativeCountry Hungary                     6.076e-01  6.697e-01   0.907 0.364310
## NativeCountry India                      -1.956e-01  2.975e-01  -0.658 0.510769
## NativeCountry Iran                       -2.249e-01  4.316e-01  -0.521 0.602249
## NativeCountry Ireland                     1.019e+00  5.315e-01   1.918 0.055083
## NativeCountry Italy                       4.964e-01  3.144e-01   1.579 0.114375
## NativeCountry Jamaica                     3.283e-01  4.190e-01   0.783 0.433368
## NativeCountry Japan                       2.463e-01  3.853e-01   0.639 0.522738
## NativeCountry Laos                       -6.498e-01  8.527e-01  -0.762 0.446026
## NativeCountry Mexico                     -7.399e-01  2.428e-01  -3.048 0.002306
## NativeCountry Nicaragua                  -7.363e-01  7.844e-01  -0.939 0.347903
## NativeCountry Outlying-US(Guam-USVI-etc) -1.104e+01  1.164e+02  -0.095 0.924436
## NativeCountry Peru                       -8.164e-01  6.229e-01  -1.311 0.189958
## NativeCountry Philippines                 3.476e-01  2.608e-01   1.333 0.182574
## NativeCountry Poland                      3.084e-02  4.140e-01   0.074 0.940618
## NativeCountry Portugal                    8.365e-01  4.377e-01   1.911 0.056001
## NativeCountry Puerto-Rico                -1.152e-01  3.447e-01  -0.334 0.738214
## NativeCountry Scotland                    8.634e-02  7.960e-01   0.108 0.913627
## NativeCountry South                      -8.590e-01  4.298e-01  -1.999 0.045628
## NativeCountry Taiwan                      7.372e-02  4.633e-01   0.159 0.873559
```

```
## NativeCountry Thailand                    -5.877e-01  6.997e-01  -0.840 0.400986
## NativeCountry Trinadad&Tobago             -1.858e+00  1.113e+00  -1.670 0.094847
## NativeCountry United-States                1.727e-01  1.258e-01   1.373 0.169900
## NativeCountry Vietnam                     -1.209e+00  5.997e-01  -2.016 0.043784
## NativeCountry Yugoslavia                   5.376e-01  6.587e-01   0.816 0.414380
##
## (Intercept)                               ***
## Age                                       ***
## WorkClass Federal-gov                     ***
## WorkClass Local-gov                       ***
## WorkClass Never-worked
## WorkClass Private                         ***
## WorkClass Self-emp-inc                    ***
## WorkClass Self-emp-not-inc
## WorkClass State-gov                       *
## WorkClass Without-pay
## Education 11th
## Education 12th
## Education 1st-4th                         .
## Education 5th-6th
## Education 7th-8th                         **
## Education 9th
## Education Assoc-acdm                      ***
## Education Assoc-voc                       ***
## Education Bachelors                       ***
## Education Doctorate                       ***
## Education HS-grad                         ***
## Education Masters                         ***
## Education Preschool
## Education Prof-school                     ***
## Education Some-college                    ***
## YearsEdu
## `Marital-Status` Married-AF-spouse        ***
## `Marital-Status` Married-civ-spouse       ***
## `Marital-Status` Married-spouse-absent
## `Marital-Status` Never-married            ***
## `Marital-Status` Separated
## `Marital-Status` Widowed
## Job Adm-clerical
## Job Armed-Forces
## Job Craft-repair                          .
## Job Exec-managerial                       ***
## Job Farming-fishing                       ***
## Job Handlers-cleaners                     ***
## Job Machine-op-inspct                     *
## Job Other-service                         ***
## Job Priv-house-serv                       *
## Job Prof-specialty                        ***
## Job Protective-serv                       ***
## Job Sales                                 ***
## Job Tech-support                          ***
## Job Transport-moving
## Relationship Not-in-family                *
## Relationship Other-relative               *
```

```
## Relationship Own-child                  *
## Relationship Unmarried
## Relationship Wife                       ***
## Race Asian-Pac-Islander                 **
## Race Black                              .
## Race Other
## Race White                              **
## Sex Male                                ***
## CapitalGain                             ***
## CapitalLoss                             ***
## HoursWorked                             ***
## NativeCountry Cambodia
## NativeCountry Canada                    *
## NativeCountry China
## NativeCountry Columbia                  **
## NativeCountry Cuba
## NativeCountry Dominican-Republic        *
## NativeCountry Ecuador
## NativeCountry El-Salvador
## NativeCountry England
## NativeCountry France                    .
## NativeCountry Germany
## NativeCountry Greece
## NativeCountry Guatemala
## NativeCountry Haiti
## NativeCountry Holand-Netherlands
## NativeCountry Honduras
## NativeCountry Hong
## NativeCountry Hungary
## NativeCountry India
## NativeCountry Iran
## NativeCountry Ireland                   .
## NativeCountry Italy
## NativeCountry Jamaica
## NativeCountry Japan
## NativeCountry Laos
## NativeCountry Mexico                    **
## NativeCountry Nicaragua
## NativeCountry Outlying-US(Guam-USVI-etc)
## NativeCountry Peru
## NativeCountry Philippines
## NativeCountry Poland
## NativeCountry Portugal                  .
## NativeCountry Puerto-Rico
## NativeCountry Scotland
## NativeCountry South                     *
## NativeCountry Taiwan
## NativeCountry Thailand
## NativeCountry Trinadad&Tobago           .
## NativeCountry United-States
## NativeCountry Vietnam                   *
## NativeCountry Yugoslavia
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 42918  on 39072  degrees of freedom
## Residual deviance: 24605  on 38975  degrees of freedom
## AIC: 24801
##
## Number of Fisher Scoring iterations: 12
```

Ah! Well we sure do get to get to view the impact of every level (dummy variable) on the output model. Before analyzing the coefficients predicted by the model, I want to examine which attributes were better for the model as compared to others.

Explanation The data produced by the model is the coefficients of each predictor. The coefficient represents the effect the value of the predictor has on our target. If we have a positive coefficient like age, as age goes up we can expect the probability of our target (IncomeClass) to go up. The final model then considers each of these coefficients in it's prediction. Different parts of the data are: - Deviance Residuals: - The Null Deviance: - Residual Deviance: - Degrees of Freedom: - AIC: - Fisher Scoring Iterations: - Standard Error: - Z Value: - P Value:

Looking at P-Values A coefficient estimate's p-values can tell us which features are valuable predictors. However, because the data is mostly qualitative, each level of each factor has a different impact on the data.

WorkClass seems like it is a good predictor *overall*, but if a given person's WorkClass is Never-worked, well the p-value is huge! Now, obviously if you have never worked your income isn't going to be very high, and the model estimates a high negative correlation. Yet the P-Value is super high!

This could be due to a number of factors: - The sample size of people who have never worked in this data is much smaller than the total population. - Our target factor is skewed, so this predictor can't differ too much from the null hypothesis - People who have never worked have varying life experiences, so the final accuracy of their coefficients isn't going to be able to fit the data

As humans we can see the this coefficient should be significant, so perhaps this isn't the best dataset for logistic regression. The summary of the model basically is this:

> While factors that you would expect to negatively impact income class do have large negative coefficents their p-values are very large because the overall target is very skewed (probably) towards what they are predicting (low income).

Probability Warning Another issue with the data is the warning:

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

This error occurs when our model fits the data so well it is most likely too perfect. This means there is *somewhat likely* an error in our data. We can check it by looking at a couple predictions:

```r
head(predict(glm1, train, type="response"), 30) # Looking at some probabilities
```

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

```
##        8990       37354       36204         116        6500       34793
## 0.006424726 0.605952839 0.445285902 0.010004641 0.015518725 0.853156276
##       19144       18326       43615       23313        9036       12545
## 0.007332188 0.511504504 0.124204561 0.695163031 0.300158154 0.065823211
##       34434       31955       21264       38205       15600       38961
## 0.312182289 0.021326621 0.002884397 0.001019163 0.001091599 0.065227546
##       29091       28084       25720       28628        9546       42563
## 0.061299106 0.414569135 0.141364327 0.069965559 0.075764004 0.796325463
##       46904       12336       43124       41030        2830       45422
## 0.001036495 0.004642817 0.018706117 0.023555186 0.212605813 0.044118244
```

Looking at just 30 fitted probabilities we see that not every single probability is 1 or 0, but another warning:

> Warning: prediction from a rank-deficent fit may be misleading

This means our number of linearly independent columns does not equal the number of parameters. Funny enough, the actual model throws out what it believes are perfectly colinear variables, causing this warning. The solution would then be to remove the colinear attributes, which will be done in just a moment.

Initial Impressions Dismissing those issues, good predictors are: - Age - Work Class - Education (Specifically higher education) - Job - Marriage Status - Sex - Hours Worked

This model makes me wonder what would happen if we selected a sample from this dataset that is less skewed, but I'm unsure what this would do to the accuracy of this model in the real world.

Improving the Model We wanted to see if removing predictors would help the overall accuracy, especially given that our predictors are somewhat dependent on each other. A brief search revealed that the anova function can show how adding each predictor effects the model.

```r
anova(glm1, test="Chisq")
```

```
## 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
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: IncomeClass
##
## Terms added sequentially (first to last)
##
##
##                  Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                             39072      42918
## Age               1   2058.2     39071      40859 < 2.2e-16 ***
## WorkClass         8    960.6     39063      39899 < 2.2e-16 ***
## Education        15   4435.2     39048      35464 < 2.2e-16 ***
```

```
## YearsEdu           0      0.0     39048      35464
## `Marital-Status`   6   6601.6     39042      28862 < 2.2e-16 ***
## Job               13    903.5     39029      27959 < 2.2e-16 ***
## Relationship       5    226.6     39024      27732 < 2.2e-16 ***
## Race               4     16.7     39020      27715  0.002227 **
## Sex                1    143.8     39019      27571 < 2.2e-16 ***
## CapitalGain        1   2083.3     39018      25488 < 2.2e-16 ***
## CapitalLoss        1    400.7     39017      25087 < 2.2e-16 ***
## HoursWorked        1    375.0     39016      24713 < 2.2e-16 ***
## NativeCountry     41    107.4     38975      24605  7.35e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Looking below, we can tell that each addition to the model is statistically relevant

Conclusions There are issues with the data, mostly a high bias and a skewed target variable, but our current model still could give good predictions given a similar data set. If you took another sample of census data just after this one it could probably predict income class a bit

Naive Bayes Model

```r
library(e1071)
nb1 <- naiveBayes(train$IncomeClass~., data=train)
nb1
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##     <=50k      >50k
## 0.7616257 0.2383743
##
## Conditional probabilities:
##        Age
## Y           [,1]     [,2]
##   <=50k 36.85352 14.08442
##   >50k  44.32006 10.59541
##
##        WorkClass
## Y                  ?  Federal-gov    Local-gov  Never-worked      Private
##   <=50k 0.0666353036 0.0239927417 0.0602842837  0.0003024295 0.7149769818
##   >50k  0.0237277217 0.0477775392 0.0811681340  0.0000000000 0.6307708825
##        WorkClass
## Y         Self-emp-inc  Self-emp-not-inc    State-gov  Without-pay
##   <=50k  0.0203971908      0.0744312645 0.0384085487 0.0005712558
##   >50k   0.0795576551      0.0929783122 0.0438050247 0.0002147305
##
```

```
##       Education
## Y              10th         11th         12th      1st-4th      5th-6th
##   <=50k 0.0343761551 0.0461373030 0.0158943513 0.0066870527 0.0126684364
##   >50k  0.0080523942 0.0079450290 0.0042946103 0.0006441915 0.0024694009
##       Education
## Y           7th-8th          9th   Assoc-acdm    Assoc-voc    Bachelors
##   <=50k 0.0243623778 0.0191874727 0.0317887026 0.0412984307 0.1266843644
##   >50k  0.0051535323 0.0035430535 0.0353231694 0.0443418510 0.2807601460
##       Education
## Y         Doctorate      HS-grad      Masters    Preschool  Prof-school
##   <=50k 0.0043684264 0.3582445647 0.0330320239 0.0021506099 0.0060149871
##   >50k  0.0372557440 0.2117242860 0.1283014816 0.0001073653 0.0531458020
##       Education
## Y       Some-college
##   <=50k  0.2371047414
##   >50k   0.1769379429
##
##       YearsEdu
## Y          [,1]     [,2]
##   <=50k  9.60375 2.440222
##   >50k  11.61134 2.390320
##
##       Marital-Status
## Y          Divorced  Married-AF-spouse  Married-civ-spouse
##   <=50k 0.1612957425       0.0007392722        0.3317315770
##   >50k  0.0544341851       0.0011810178        0.8576336697
##       Marital-Status
## Y       Married-spouse-absent  Never-married    Separated      Widowed
##   <=50k          0.0156927316   0.4129507040 0.0392486307 0.0383413421
##   >50k           0.0042946103   0.0637749624 0.0080523942 0.0106291604
##
##       Job
## Y                  ?  Adm-clerical  Armed-Forces  Craft-repair  Exec-managerial
##   <=50k 0.0669377331  0.1311199973  0.0002688262  0.1280620989     0.0846130582
##   >50k  0.0237277217  0.0665664591  0.0004294610  0.1177796865     0.2450075156
##       Job
## Y       Farming-fishing  Handlers-cleaners  Machine-op-inspct  Other-service
##   <=50k     0.0360899224       0.0515138277       0.0709701267   0.1275580497
##   >50k      0.0147090402       0.0118101782       0.0299549066   0.0180373631
##       Job
## Y       Priv-house-serv  Prof-specialty  Protective-serv        Sales
##   <=50k     0.0064518297    0.0912665076     0.0182465809 0.1083369737
##   >50k      0.0001073653    0.2427528452     0.0274855057 0.1247584282
##       Job
## Y       Tech-support  Transport-moving
##   <=50k  0.0285627877      0.0500016802
##   >50k   0.0357526304      0.0411208933
##
##       Relationship
## Y          Husband  Not-in-family  Other-relative   Own-child   Unmarried
##   <=50k 0.291138815    0.304916160     0.038475755 0.201082026 0.131187204
##   >50k  0.758213442    0.106935796     0.004401976 0.009448143 0.025338201
##       Relationship
## Y             Wife
```

```
##   <=50k 0.033200040
##   >50k  0.095662444
##
##        Race
## Y         Amer-Indian-Eskimo  Asian-Pac-Islander       Black       Other
##   <=50k         0.011223495         0.029134043 0.110151551 0.009509728
##   >50k          0.004831437         0.034571613 0.049280653 0.004187245
##        Race
## Y             White
##   <=50k 0.839981182
##   >50k  0.907129053
##
##        Sex
## Y          Female      Male
##   <=50k 0.3903693 0.6096307
##   >50k  0.1526734 0.8473266
##
##        CapitalGain
## Y            [,1]       [,2]
##   <=50k  150.5972   970.6327
##   >50k  3944.5582 14468.3342
##
##        CapitalLoss
## Y            [,1]     [,2]
##   <=50k  53.72828 310.1593
##   >50k  194.93172 595.5030
##
##        HoursWorked
## Y           [,1]     [,2]
##   <=50k 38.85846 12.38484
##   >50k  45.41207 11.17745
##
##        NativeCountry
## Y                  ?     Cambodia       Canada        China     Columbia
##   <=50k 1.717128e-02 6.048590e-04 3.125105e-03 2.385833e-03 2.385833e-03
##   >50k  1.900365e-02 6.441915e-04 5.475628e-03 3.113592e-03 3.220958e-04
##        NativeCountry
## Y               Cuba  Dominican-Republic      Ecuador  El-Salvador      England
##   <=50k 2.654659e-03        2.822675e-03 1.075305e-03 3.998790e-03 2.318626e-03
##   >50k  2.684131e-03        2.147305e-04 5.368263e-04 9.662873e-04 4.294610e-03
##        NativeCountry
## Y             France      Germany       Greece    Guatemala        Haiti
##   <=50k 4.368426e-04 3.931584e-03 9.408918e-04 2.385833e-03 1.881784e-03
##   >50k  1.395748e-03 5.046167e-03 1.181018e-03 3.220958e-04 6.441915e-04
##        NativeCountry
## Y         Holand-Netherlands     Honduras         Hong      Hungary        India
##   <=50k         3.360328e-05 4.032394e-04 6.720656e-04 3.024295e-04 2.385833e-03
##   >50k          0.000000e+00 2.147305e-04 5.368263e-04 5.368263e-04 5.690359e-03
##        NativeCountry
## Y               Iran      Ireland        Italy      Jamaica        Japan
##   <=50k 1.108908e-03 7.056689e-04 2.117007e-03 2.318626e-03 1.512148e-03
##   >50k  1.717844e-03 9.662873e-04 3.113592e-03 1.503114e-03 2.898862e-03
##        NativeCountry
## Y               Laos       Mexico    Nicaragua  Outlying-US(Guam-USVI-etc)
```

```
##   <=50k 5.040492e-04 2.459760e-02 1.310528e-03                5.712558e-04
##   >50k  2.147305e-04 3.650419e-03 2.147305e-04                0.000000e+00
##        NativeCountry
## Y                Peru  Philippines       Poland     Portugal  Puerto-Rico
##   <=50k 1.142512e-03 5.342921e-03 1.680164e-03 1.377734e-03 4.267617e-03
##   >50k  4.294610e-04 7.408203e-03 1.288383e-03 1.073653e-03 1.825209e-03
##        NativeCountry
## Y            Scotland        South       Taiwan     Thailand  Trinadad&Tobago
##   <=50k 3.696361e-04 2.285023e-03 1.075305e-03 6.720656e-04     6.384623e-04
##   >50k  3.220958e-04 1.717844e-03 2.254670e-03 5.368263e-04     1.073653e-04
##        NativeCountry
## Y        United-States      Vietnam   Yugoslavia
##   <=50k   8.921671e-01 1.915387e-03 4.032394e-04
##   >50k    9.147520e-01 5.368263e-04 6.441915e-04
```

Naive Bayes produces a model that first finds the prior probability (A-priori, or the probability of having <=50k or >50k with no considerations of other data) and then finds the probability of the income given each condition independently. For example the table for Sex states that the probability that someone is female given that you make less than 50k is ~40%, while if a person makes more than 50k the chance they are a woman is ~15%.

We also see the results for quantified predictors. For a continuous predictor like age, the mean age for people <=50k is 36.85352 while people >50k are older at a mean of 44.32006 years old.

The model may just be finding the independent probabilities of the target event given each predictor but using all of the probabilities at once can provide a pretty good guess. Good enough to predict our training data!

Issues in the Data It's worth noting once again that our predictors may not be completely independent but our model here assumes they are. That is why we call it naive! With such a large amount of data, probability can overcome the shortcomings of this assumption and we could get reasonably accurate predictions

Predictions

```
p1 <- predict(glm1, 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
```

```
pred1 <- ifelse(p1>0.5, ">50k", "<=50k")
head(pred1)
```

```
##      20      23      25      62      68      77
##  ">50k" "<=50k"  ">50k" "<=50k" "<=50k" "<=50k"
```

```
head(test$IncomeClass)
```

```
## [1] >50k  <=50k <=50k <=50k <=50k <=50k
## Levels: <=50k >50k
```

```r
cm1 <- caret::confusionMatrix(as.factor(pred1), reference=test$IncomeClass)
cm1
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction <=50k >50k
##      <=50k  6869  979
##      >50k    527 1394
##
##                Accuracy : 0.8458
##                  95% CI : (0.8385, 0.8529)
##     No Information Rate : 0.7571
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5519
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9287
##             Specificity : 0.5874
##          Pos Pred Value : 0.8753
##          Neg Pred Value : 0.7257
##              Prevalence : 0.7571
##          Detection Rate : 0.7031
##    Detection Prevalence : 0.8034
##       Balanced Accuracy : 0.7581
##
##        'Positive' Class : <=50k
##
```

```r
p2 <- predict(nb1, newdata=test, type="class")
head(p2)
```

```
## [1] >50k  <=50k <=50k <=50k <=50k <=50k
## Levels: <=50k >50k
```

```r
head(test$IncomeClass)
```

```
## [1] >50k  <=50k <=50k <=50k <=50k <=50k
## Levels: <=50k >50k
```

```r
cm2 <- caret::confusionMatrix(as.factor(p2), test$IncomeClass)
cm2
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction <=50k >50k
##      <=50k  6898 1225
##      >50k    498 1148
```

```
##
##                Accuracy : 0.8236
##                  95% CI : (0.8159, 0.8311)
##     No Information Rate : 0.7571
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.4648
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9327
##             Specificity : 0.4838
##          Pos Pred Value : 0.8492
##          Neg Pred Value : 0.6974
##              Prevalence : 0.7571
##          Detection Rate : 0.7061
##    Detection Prevalence : 0.8315
##       Balanced Accuracy : 0.7082
##
##        'Positive' Class : <=50k
##
```

```
cm1$byClass
```

```
##          Sensitivity          Specificity       Pos Pred Value
##            0.9287453            0.5874421            0.8752548
##       Neg Pred Value            Precision               Recall
##            0.7256637            0.8752548            0.9287453
##                   F1           Prevalence       Detection Rate
##            0.9012070            0.7570888            0.7031426
## Detection Prevalence    Balanced Accuracy
##            0.8033576            0.7580937
```

Initial conclusion The initial conclusion to be drawn from our predictions is that our accuracy for both our models is okay, and our logistic regression model did better than our Naive Bayes. This could probably be due to Naive Bayes often doing better with small data sets while logistic regression works better with large datasets. On the other hand the logistic regression model might have still been overwhelmed by the amount of factors, and the accuracy was only ~84%.

The confusion matrix tells us True Positive, False Positive, True Negative, and False Negative results from applying the model to the test data. We can use the ratios between these numbers to evaluate useful metrics like accuracy or sensitivity.

The Confusion Matrix

```
          Reference
Prediction <=50k >50k
     <=50k  6898 1225
     >50k    498 1148
```

Just for an example we are looking at the naive bayes confusion matrix. - 6898: The number of True Positives - 1148: The number of True Negatives - 498: The number of False Negatives - 1225: The number of False Positives

We can use these to calculate other metrics

Accuracy

```
Logistic R.: ~85%
Naive Bayes: ~82%
```

The diagonals, or our true results, divided by all of our predictions is our accuracy, or the percentage we were correct. As you can see, our logistic regression model was accurate more of the time. Most likely because it thrived more with the large amount of data.

Sensitivity & Specificity

```
Logistic R.: 0.9287 and 0.5874
Naive Bayes: 0.9327 and 0.4838
```

Naive Bayes had a higher sensitivity, which is the number of true positives out of true positives + false negatives (the number of positives in the data). If we were trying to perhaps locate all people with "low" income but didn't care about our accuracy with people above 50k, the stat shows naive bayes could be useful.

Specificity is the measure of true negatives in the negative class. We can tell then that we were much better at identifying our people with <=50k income than people with >50k income. However, logistic regression was still better than Naive Bayes in this stat.

Well you ignore part of the data and perhaps get to ignore issues ini your model (like ignoring a bunch of false negatives), these are great for getting what matters out of data.

Kappa

```
Logistic R.: 0.5519
Naive Bayes: 0.4648
```

Woah! These aren't the best numbers, but considering this is a measure of accuracy that corrects for prediction by chance, I'm surprised the number is so high. The data set was skewed, it seemed a large margin of the success of our models was due to random chance. According to a reference on kappa scores though, these numbers are in "moderate agreement" with what is expected.

Kappa is great for regarding datasets where the random chance of getting a prediction high is right. Of course, there isn't a consensus on what the number means on a scale, but its still generally useful.

```r
library(ROCR)
head(p1)
```

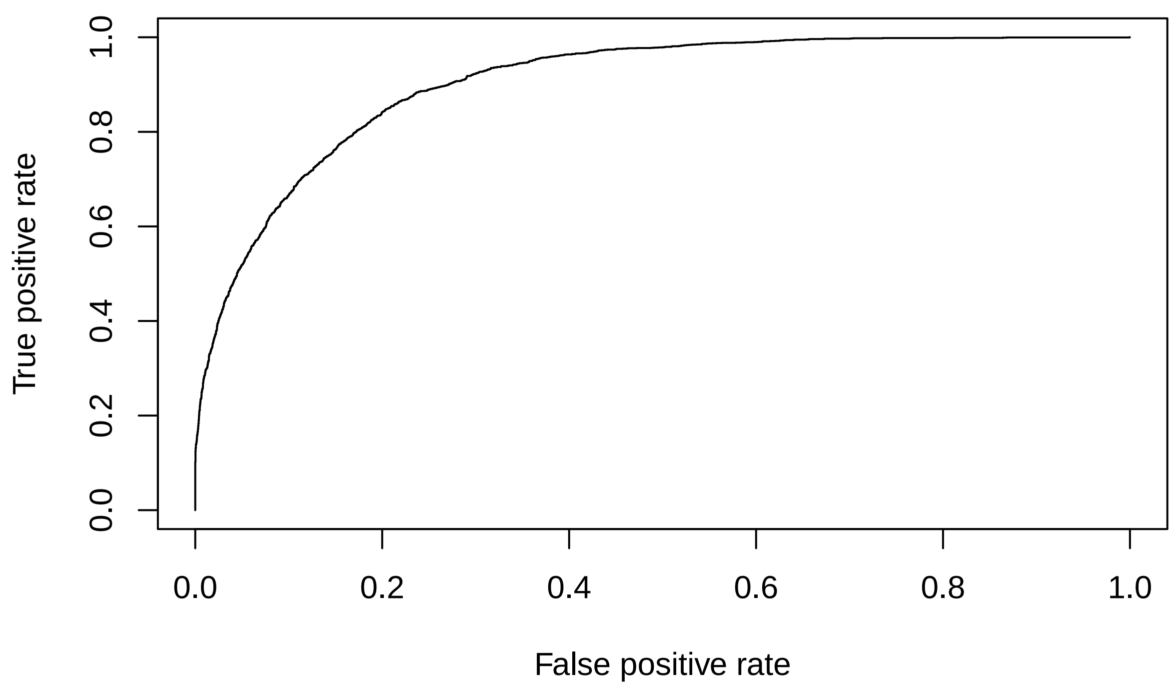
ROC Curves and AUC

```
##          20          23          25          62          68          77
## 0.820678098 0.002768888 0.532388392 0.083253034 0.002496240 0.232514435
```

```r
head(test$IncomeClass)
```

```
## [1] >50k  <=50k <=50k <=50k <=50k <=50k
## Levels: <=50k >50k
```

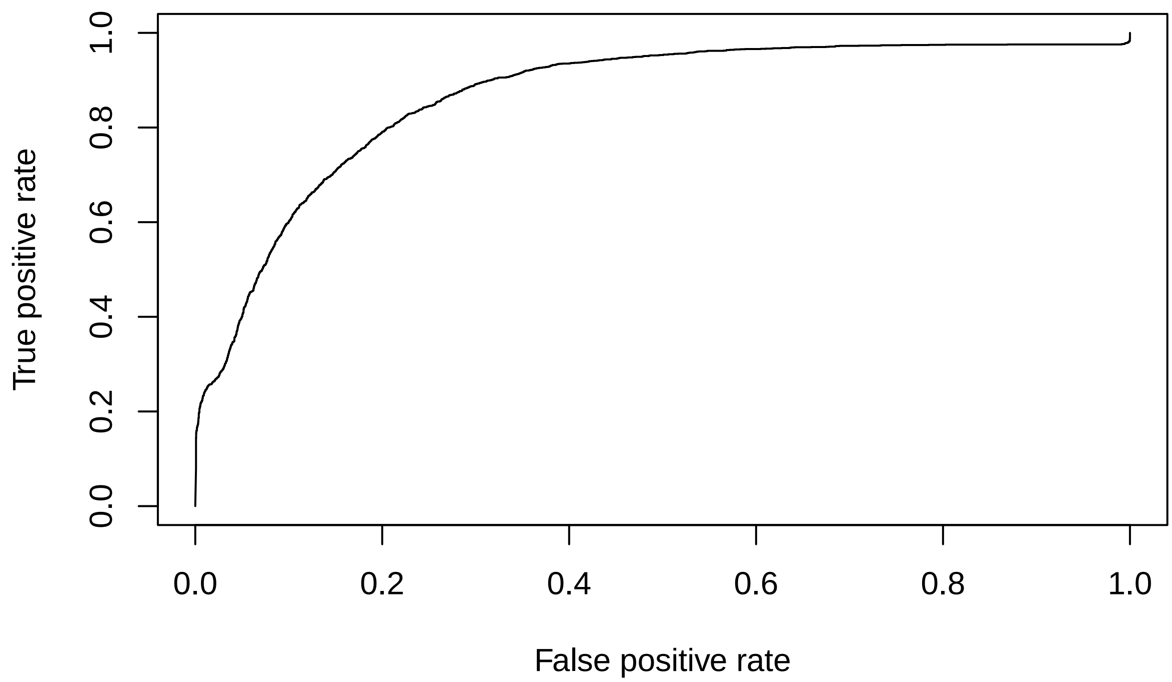
```r
pr <- prediction(p1, test$IncomeClass)
prf <- performance(pr, measure = "tpr", x.measure = "fpr")
plot(prf)
```




```r
# Compute AUC
auc <- performance(pr, measure = "auc")
auc <- auc@y.values[[1]]
auc
```

```
## [1] 0.9031762
```

```r
library(ROCR)
p2raw <- predict(nb1, newdata=test, type="raw")[,2]
pr2 <- prediction(p2raw,as.numeric(test$IncomeClass))
prf2 <- performance(pr2, measure = "tpr", x.measure = "fpr")
plot(prf2)
```




```
# Compute AUC
auc <- performance(pr2, measure = "auc")
auc <- auc@y.values[[1]]
auc
```

```
## [1] 0.8662319
```

```
# Logistic Regression
mltools::mcc(as.factor(pred1), test$IncomeClass)
```

Matthew's Correlation Coefficient (MCC):

```
## [1] 0.5569439
```

```
# Naive Bayes
mltools::mcc(p2, test$IncomeClass)
```

```
## [1] 0.4771213
```

.4771213 is smack dab in between a perfect model (1) and a model that is perfectly average (0). Pretty good!

Strengths and Weaknesses

Logistic Regression basically is attempting to draw a line between classes. It ends up being quite computationally inexpensive, easy to understand, and does its job well if classes are easy to separate. But because of it's simplicity as a line, it just isn't complex enough to capture complex non-linear decision boundaries. Naive Bayes is also simple, but with the added bonus that it works well with high dimensions (complex data sets) *if* they aren't too big. It's simple however because it assumes variables are independent, and ends up lacking with larger data sets.

Summary of Metrics

Accuracy being the ratio of correct predictions to incorrect predictions, it is broadly useful. But often we are searching for subsections of accuracy. Sensitivity is good for detecting the amount we get one (the positive) class and ignores the other. Specificity on the other hand is the ratio of correct negative classes. This means we can use these metrics to see how ell our data is at guessing what matters in the at. If we want to see general accuracy, but account for the chance of getting the prediction randomly correct, Kappa is great for checking that.

Now ROC... well it graphs the true positive rate and the false positive rate (sensitivity and specificity). Unfortunately we tried til the deadline to get this to work for Naive Bayes but we swear we understand what it means! The name, Receiver Operator Characteristic curve comes from signal detection theory so it doesn't help much to remind what it means. However, basically it graphs the trade off of a model between sensitivity and specificity. The Area under the curve then represents how much the model is capable of distinguishing between classes.

The MCC is a metric that basically gives a good value if you get a good reliable rate in all 4 values of the confusion matrix. The values are considered in proportional the size of the positive and negative values. Rather then combining the sensitivity and specificity of a metric into a single metric (like with an F1-Score), MCC considers the size of of negative samples. MCC's account for class distribution makes it great at providing an accuracy rating for the whole model rated from -1 to 1.

Conclusion

We have 1 large takeaway from this data, linear data has limitations, and none of that is helped by having a skewed data set. In the future we would like to select a data set that has less of skewed target, or at least try to sample this data at a better ratio again. It was fun to look at though!