# Data622 - Group2 - FinalProject

Adam Gersowitz, Amit Kapoor, Kevin Potter, Zachary Palmore, Paul Perez

11/26/2021

## Contents



# Overview

As income inequality grows throughout the world, understanding the relationships between an individuals income and the other factors in this study we can better identify and address the underlying causes for the inequalities. This study will analyze how 15 factors such as age, county, working class, sex, race, education and more influence our target variable, income from a diverse dataset of over 48,842 individuals. The goal of this analysis is to develop models that best predict income, so that these models can be used to make better decisions when considering income from occupations.

The dataset for this project is from the UCI Machine Learning Repository. This dataset was donated by Ron Kohavi and Barry Becker. Extraction was done by Barry Becker from the 1994 Census database. A set of reasonably clean records was extracted using the following conditions: ((AAGE>16) && (AGI>100) && (AFNLWGT>1)&& (HRSWK>0)). Relevant paper is from Ron Kohavi, "Scaling Up the Accuracy of

Naive-Bayes Classifiers: a Decision-Tree Hybrid", Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 1996. Our relevant prediction task based on this peer-reviewed paper is to determine whether a person makes over 50K a year.

# Approach

In this analysis we attempt to predict the income of individuals given a host of variables. For purposes of simplicity, our target income variable has been broken into two categories: greater than $50,000 or less than or equal to $50,000. Ideally, this will makes its prediction less prone to error since we are using a host of categorical variables. We then explore the relationships between our variables, make any necessary changes prior to modeling, develop several models, and evaluate them using confusions matricies. Our focus in this analysis, will be on identifying the variables that improve real-world accuracy to best capture the full context of income interactions with our variables.

We begin with data exploration to understand the relationships our target variable 'Income' will have with our variables and the variables' relationships to each other. This allows us to determine the steps necessary to set up for model development. Once we have an understanding of these variables we use that knowledge to prepare the data. We handle missing values, subset, train and split the data 75/25 so that we may better extract information when modeling. Then, we build the models and predict with the testing dataset. We focus on prediction accuracy when assessing the models but consider a host of performance statistics and real-world applications to determine which model is best.

# Data Exploration

In this section we begin exploring the data by creating a table with the variable names' and descriptions. We first identify the characteristics of the data to help with properly labeling and categorizing our factors. To better understand the data, we then summarize those characteristics and present them in the data summary section. Additional visualizations and correlations are created to discover any unseen patterns or potentially problematic areas prior to preparing the data.

## Data Characteristics

There are 48842 observations of 15 variables. Each observation is for individual's income data with it's corresponding variables of interest. Below is the description of the variables of interest in the data set.

| VARIABLE NAME | DESCRIPTION |
|---|---|
| age | continuous |
| workclass | Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked |
| fnlwgt | continuous |
| education | Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool |
| education-num | continuous |
| marital-status | Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse |
| occupation | Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces |
| relationship | Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried |
| race | White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black |
| sex | Female, Male |

| VARIABLE NAME | DESCRIPTION |
|---|---|
| capital-gain | continuous |
| capital-loss | continuous |
| hours-per-week | continuous |
| native-country | United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands |
| income | >50K, <=50K |

We continue by identifying any potential sources of error in the analysis. This includes calculating the number of missing and outlier values for each variable, determining and assigning the proper methods to reduce missing and problematic data points. Due to the unique dataset, we determing and convert the variables to the proper data types. Here again, for simplicity and the most effective use of the data, all are converted to factors with multiple levels as intended by the orignial data contributer. A gridded table with frequencies of each factor's level, its identity, and the number of valid or missing data points are tabulated and presented below.

## Data summary

Below is a summary of the census income dataset. For this process we have already adjusted the data types to their proper forms. This summarizing function quantifies each variable in a manner consistent with their types. We notice the levels of each factor in the 'Stats/Values' column, the frequency of valid (non-missing) observations per level of our factors, and the quantity and percent missing alongside them. We review these statistics to identify any issues with each variable.

```
## Data Frame Summary
## income_data
## Dimensions: 48842 x 15
## Duplicates: 52
##
## +----+-----------------+--------------------------------+----------------------+---------+---------
## | No | Variable        | Stats / Values                 | Freqs (% of Valid)   | Valid   | Missing
## +====+=================+================================+======================+=========+=========
## | 1  | age             | Mean (sd) : 38.6 (13.7)        | 74 distinct values   | 48842   | 0
## |    | [integer]       | min < med < max:               |                      | (100.0%) | (0.0%)
## |    |                 | 17 < 37 < 90                   |                      |         |
## |    |                 | IQR (CV) : 20 (0.4)            |                      |         |
## +----+-----------------+--------------------------------+----------------------+---------+---------
## | 2  | workclass       | 1. ?                           |  2799 ( 5.7%)        | 48842   | 0
## |    | [factor]        | 2. Federal-gov                 |  1432 ( 2.9%)        | (100.0%) | (0.0%)
## |    |                 | 3. Local-gov                   |  3136 ( 6.4%)        |         |
## |    |                 | 4. Never-worked                |    10 ( 0.0%)        |         |
## |    |                 | 5. Private                     | 33906 (69.4%)        |         |
## |    |                 | 6. Self-emp-inc                |  1695 ( 3.5%)        |         |
## |    |                 | 7. Self-emp-not-inc            |  3862 ( 7.9%)        |         |
## |    |                 | 8. State-gov                   |  1981 ( 4.1%)        |         |
## |    |                 | 9. Without-pay                 |    21 ( 0.0%)        |         |
## +----+-----------------+--------------------------------+----------------------+---------+---------
## | 3  | fnlwgt          | Mean (sd) : 189664.1 (105604)  | 28523 distinct values | 48842   | 0
```

```
## |    | [integer]       | min < med < max:                 |                        | (100.0%) | (0.0%)
## |    |                 | 12285 < 178144.5 < 1490400       |                        |          |
## |    |                 | IQR (CV) : 120091.5 (0.6)        |                        |          |
## +----+-----------------+----------------------------------+------------------------+----------+---------
## | 4  | education       | 1. 10th                          |  1389 ( 2.8%)          | 48842    | 0
## |    | [factor]        | 2. 11th                          |  1812 ( 3.7%)          | (100.0%) | (0.0%)
## |    |                 | 3. 12th                          |   657 ( 1.3%)          |          |
## |    |                 | 4. 1st-4th                       |   247 ( 0.5%)          |          |
## |    |                 | 5. 5th-6th                       |   509 ( 1.0%)          |          |
## |    |                 | 6. 7th-8th                       |   955 ( 2.0%)          |          |
## |    |                 | 7. 9th                           |   756 ( 1.5%)          |          |
## |    |                 | 8. Assoc-acdm                    |  1601 ( 3.3%)          |          |
## |    |                 | 9. Assoc-voc                     |  2061 ( 4.2%)          |          |
## |    |                 | 10. Bachelors                    |  8025 (16.4%)          |          |
## |    |                 | [ 6 others ]                     | 30830 (63.1%)          |          |
## +----+-----------------+----------------------------------+------------------------+----------+---------
## | 5  | education_num   | Mean (sd) : 10.1 (2.6)           | 16 distinct values     | 48842    | 0
## |    | [integer]       | min < med < max:                 |                        | (100.0%) | (0.0%)
## |    |                 | 1 < 10 < 16                      |                        |          |
## |    |                 | IQR (CV) : 3 (0.3)               |                        |          |
## +----+-----------------+----------------------------------+------------------------+----------+---------
## | 6  | marital_status  | 1. Divorced                      |  6633 (13.6%)          | 48842    | 0
## |    | [factor]        | 2. Married-AF-spouse             |    37 ( 0.1%)          | (100.0%) | (0.0%)
## |    |                 | 3. Married-civ-spouse            | 22379 (45.8%)          |          |
## |    |                 | 4. Married-spouse-absent         |   628 ( 1.3%)          |          |
## |    |                 | 5. Never-married                 | 16117 (33.0%)          |          |
## |    |                 | 6. Separated                     |  1530 ( 3.1%)          |          |
## |    |                 | 7. Widowed                       |  1518 ( 3.1%)          |          |
## +----+-----------------+----------------------------------+------------------------+----------+---------
## | 7  | occupation      | 1. ?                             |  2809 ( 5.8%)          | 48842    | 0
## |    | [factor]        | 2. Adm-clerical                  |  5611 (11.5%)          | (100.0%) | (0.0%)
## |    |                 | 3. Armed-Forces                  |    15 ( 0.0%)          |          |
## |    |                 | 4. Craft-repair                  |  6112 (12.5%)          |          |
## |    |                 | 5. Exec-managerial               |  6086 (12.5%)          |          |
## |    |                 | 6. Farming-fishing               |  1490 ( 3.1%)          |          |
## |    |                 | 7. Handlers-cleaners             |  2072 ( 4.2%)          |          |
## |    |                 | 8. Machine-op-inspct             |  3022 ( 6.2%)          |          |
## |    |                 | 9. Other-service                 |  4923 (10.1%)          |          |
## |    |                 | 10. Priv-house-serv              |   242 ( 0.5%)          |          |
## |    |                 | [ 5 others ]                     | 16460 (33.7%)          |          |
## +----+-----------------+----------------------------------+------------------------+----------+---------
## | 8  | relationship    | 1. Husband                       | 19716 (40.4%)          | 48842    | 0
## |    | [factor]        | 2. Not-in-family                 | 12583 (25.8%)          | (100.0%) | (0.0%)
## |    |                 | 3. Other-relative                |  1506 ( 3.1%)          |          |
## |    |                 | 4. Own-child                     |  7581 (15.5%)          |          |
## |    |                 | 5. Unmarried                     |  5125 (10.5%)          |          |
## |    |                 | 6. Wife                          |  2331 ( 4.8%)          |          |
## +----+-----------------+----------------------------------+------------------------+----------+---------
## | 9  | race            | 1. Amer-Indian-Eskimo            |   470 ( 1.0%)          | 48842    | 0
## |    | [factor]        | 2. Asian-Pac-Islander            |  1519 ( 3.1%)          | (100.0%) | (0.0%)
## |    |                 | 3. Black                         |  4685 ( 9.6%)          |          |
## |    |                 | 4. Other                         |   406 ( 0.8%)          |          |
## |    |                 | 5. White                         | 41762 (85.5%)          |          |
## +----+-----------------+----------------------------------+------------------------+----------+---------
```

```
## | 10 | sex              | 1. Female                          | 16192 (33.2%)          | 48842    | 0
## |    | [factor]         | 2. Male                            | 32650 (66.8%)          | (100.0%) | (0.0%)
## +----+-----------------+-----------------------------------+------------------------+----------+---------
## | 11 | capital_gain     | Mean (sd) : 1079.1 (7452)          | 123 distinct values    | 48842    | 0
## |    | [integer]        | min < med < max:                   |                        | (100.0%) | (0.0%)
## |    |                  | 0 < 0 < 99999                      |                        |          |
## |    |                  | IQR (CV) : 0 (6.9)                 |                        |          |
## +----+-----------------+-----------------------------------+------------------------+----------+---------
## | 12 | capital_loss     | Mean (sd) : 87.5 (403)             | 99 distinct values     | 48842    | 0
## |    | [integer]        | min < med < max:                   |                        | (100.0%) | (0.0%)
## |    |                  | 0 < 0 < 4356                       |                        |          |
## |    |                  | IQR (CV) : 0 (4.6)                 |                        |          |
## +----+-----------------+-----------------------------------+------------------------+----------+---------
## | 13 | hours_per_week   | Mean (sd) : 40.4 (12.4)            | 96 distinct values     | 48842    | 0
## |    | [integer]        | min < med < max:                   |                        | (100.0%) | (0.0%)
## |    |                  | 1 < 40 < 99                        |                        |          |
## |    |                  | IQR (CV) : 5 (0.3)                 |                        |          |
## +----+-----------------+-----------------------------------+------------------------+----------+---------
## | 14 | native_country   | 1. ?                               |   857 ( 1.8%)          | 48842    | 0
## |    | [factor]         | 2. Cambodia                        |    28 ( 0.1%)          | (100.0%) | (0.0%)
## |    |                  | 3. Canada                          |   182 ( 0.4%)          |          |
## |    |                  | 4. China                           |   122 ( 0.2%)          |          |
## |    |                  | 5. Columbia                        |    85 ( 0.2%)          |          |
## |    |                  | 6. Cuba                            |   138 ( 0.3%)          |          |
## |    |                  | 7. Dominican-Republic              |   103 ( 0.2%)          |          |
## |    |                  | 8. Ecuador                         |    45 ( 0.1%)          |          |
## |    |                  | 9. El-Salvador                     |   155 ( 0.3%)          |          |
## |    |                  | 10. England                        |   127 ( 0.3%)          |          |
## |    |                  | [ 32 others ]                      | 47000 (96.2%)          |          |
## +----+-----------------+-----------------------------------+------------------------+----------+---------
## | 15 | income           | 1. <=50K                           | 37155 (76.1%)          | 48842    | 0
## |    | [factor]         | 2. >50K                            | 11687 (23.9%)          | (100.0%) | (0.0%)
## +----+-----------------+-----------------------------------+------------------------+----------+---------
```

At first glance, it appears none of the data are missing values now and each variables is a factor data type as we intended but we begin to notice a few issues. Certain variables contain a multitude of distinct levels and as such, are interpreted as numeric data types with statistics for mean, median, minima, maxima, standard deviation and interquartile ranges (IQR). For example in the variables `age`, `fnlwgt`, `capital_gains`, `capital_loss` and `hours_per_week` produce nearly 100+ levels each with `fnlwgt` having 28523 levels. We will need to decide if these are worth adjusting further to capture the full picture of the relationships between the variables and our target.

Following the Missing column, it seems none of the columns have missing values but Stats / Values value column shows the variables that have value as '?'. `workclass`, `occupation`, `native_country` columns have values as '?'. The proportion of values for several columns shows significant differences and skew. For example, 67% of this dataset contains males applicants based on observations of the `sex` variable and 85.5% of data points are white people given the `race` variable. Due to the disproportionate levels within the variables we should expect the data is not representative of a larger population unless that population happens to have similar proportions.

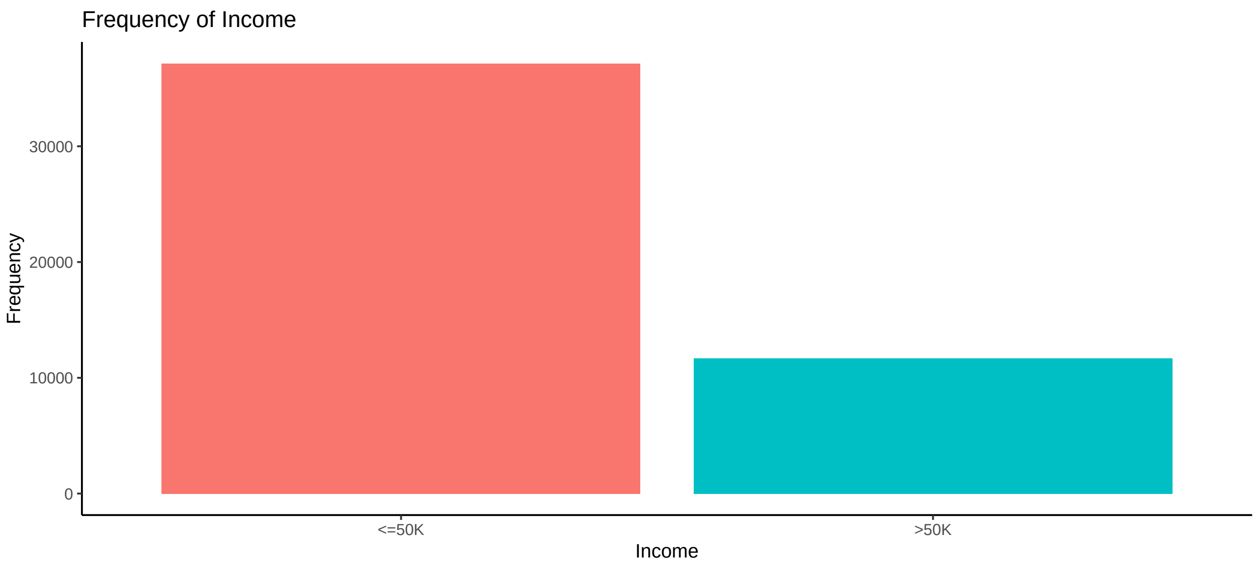
Our numeric variables `age`, `fnlwgt`, `capital_gain`, `capital_loss` show signs of skew through the differences in their mean and medians as well as their ranges. The lowest value of `fnlwgt` variable was 12285, while the highest was 1490400. A similar problem exists with variables `capital_gain` and `capital_loss.`

We consider a different summary method, which at its base function calculates those statistical parameters previously mentioned and counts the number of observations for each level as performed above. This should confirm our previous grid table results but we should also look for changes, if there are any. The results of this new summary method are shown.

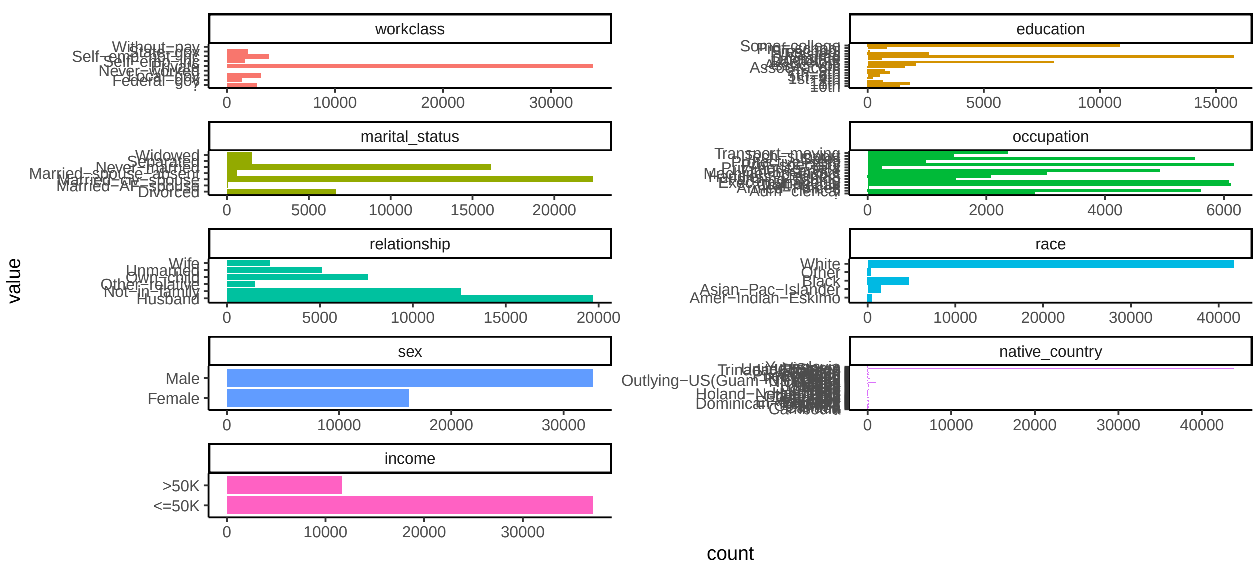
```
##      age                   workclass         fnlwgt
##  Min.   :17.00   Private         :33906   Min.   :  12285
##  1st Qu.:28.00   Self-emp-not-inc: 3862   1st Qu.: 117550
##  Median :37.00   Local-gov       : 3136   Median : 178144
##  Mean   :38.64   ?               : 2799   Mean   : 189664
##  3rd Qu.:48.00   State-gov       : 1981   3rd Qu.: 237642
##  Max.   :90.00   Self-emp-inc    : 1695   Max.   :1490400
##                  (Other)         : 1463
##         education     education_num                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
##            occupation            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         capital_gain    capital_loss    hours_per_week
##  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
##
##        native_country    income
##  United-States:43832   <=50K:37155
##  Mexico       :  951   >50K :11687
##  ?            :  857
##  Philippines  :  295
##  Germany      :  206
##  Puerto-Rico  :  184
##  (Other)      : 2517
```

With this method, our first results are confirmed. However, there appear to be few differences, if any, in these results. The only noticible change is to our target variable, income, where the previous function interpreted the values as factors without levels rather than a series of character strings as this new method did. This indicates we might not need to make any further changes to the data types or adjustments in the quantity of missing values (which might have included those with the '?' level within their factors) or outliers.

We take a closer look at our target variable to get a sense of what we are trying to predict. We also look for any inate imbalances within the target by spotting any additional unintentional biases towards a specific income level. We visualize the proportions for other factors as well to see just how skewed and disproportionate this dataset is. We include missing values as '?' to demonstrate their influence on the dataset as well. The chart below shows the distribution of all categorical variables, which includes the factors mentioned previously.

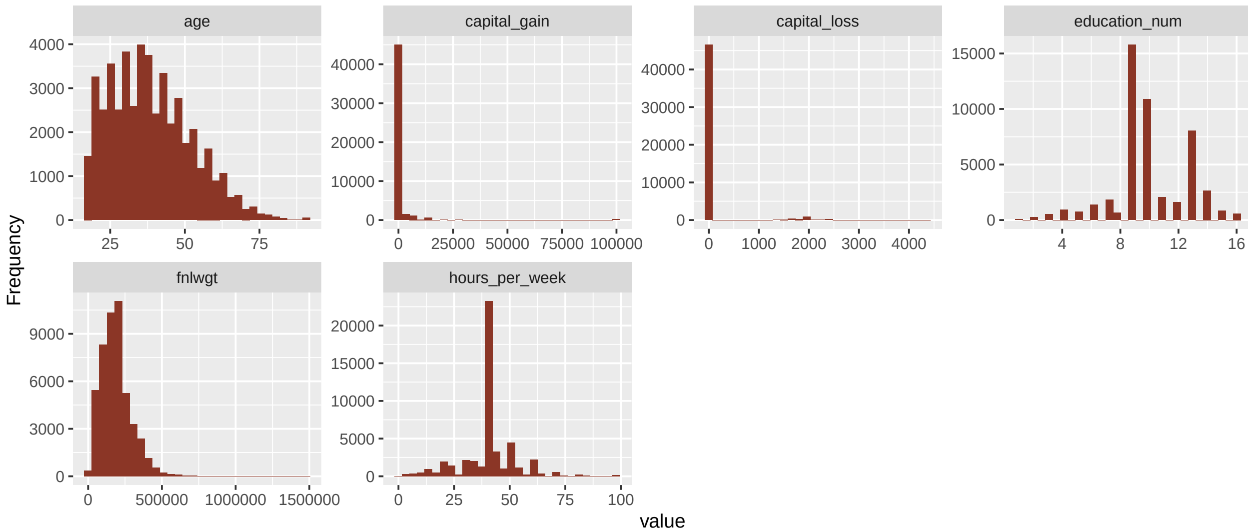



In this first bar chart we notice that there is almost triple the frequency of individuals with income less than or equal to $50,000 than the frequency of those who are above. This could be problematic for predicting the minority class (those with >50k) but we also know from the literature that this is realistic. In the population of the world, there is a lot fewer people who have more wealth than those who have less of it. For this reason, we note the disproportionate share of income towards those under less than of equal to $50,000 (<=50k) but leave it as is because this is an observation of real-world statistics.



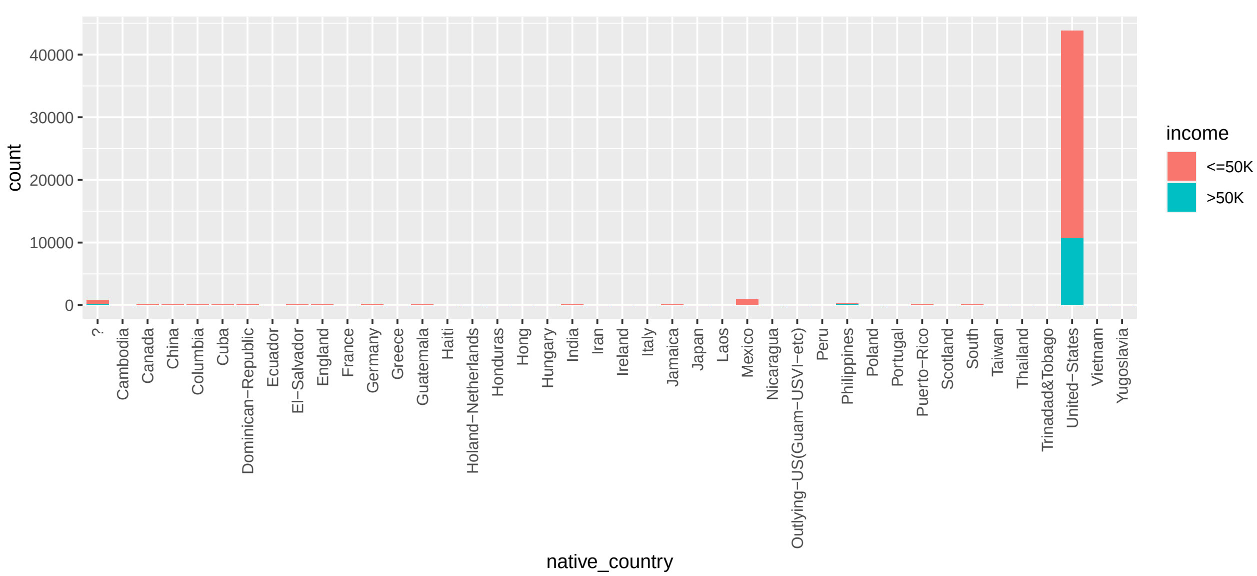

From this chart, it is very clear we have a dataset with mostly males, that are white, and in a private working class. Individuals in this survey were also mostly married as noted by the `married_civ_spouse`, `married_af_spouse`, `seperated` and `widowed` frequencies above. To us, the only case in which marriage did not occur were those who were in the `never_married` category. It is not likely that this will help us predict income since it disproportionately favors those who have been married at least once at some point in their lives. We can clearly see this growing disproportionality in the relationship variable where most individuals are husbands who also did not have kids. We see missing values in form of question mark (?) for variables `workclass`, `occupation`, `native_country`. The predictor variable Income is also shown as mostly <=50k value so we know the results agree with the previous chart. Occupation seems to have no underlying patterns or skew but we should note that individuals in U.S. dominate the respondents list. This makes understanding differences between income relationships in countries quite difficult for our purposes.

We also generate histograms with the count of each observation to assess our numeric variable distributions. This will let us know more about the skewness, average values, and where potential outliers may be found for our numeric variables. The graph below shows their distributions.

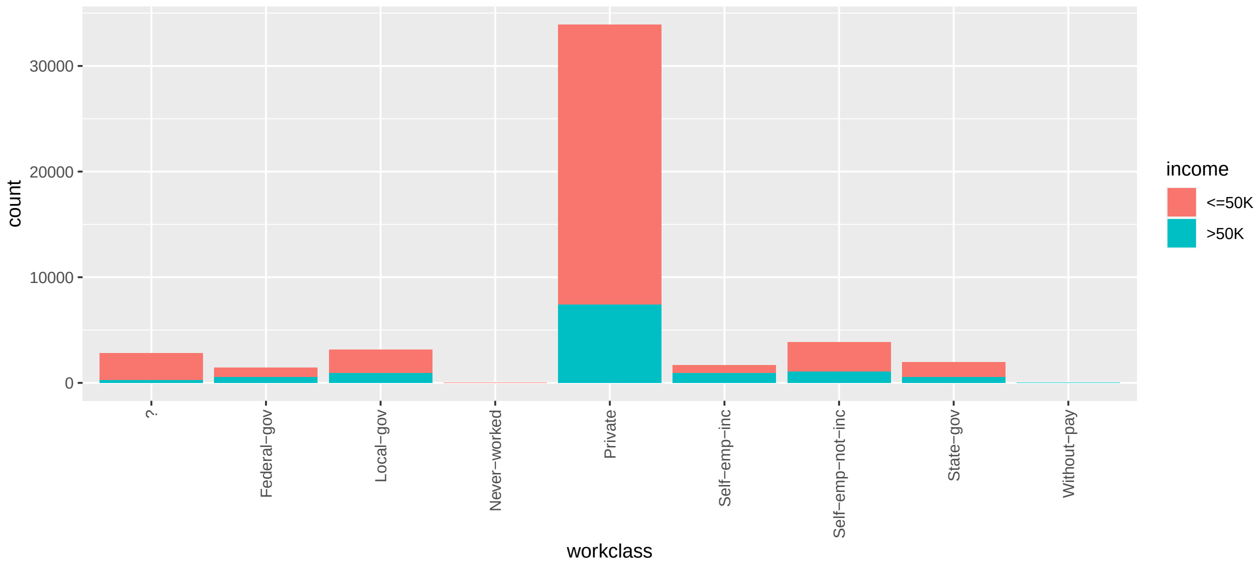



As expected, the number of hours per week spikes at about 40. This makes sense since most people in the U.S. tend to work about 40 hours per week and that is who is best represnted within this dataset. Unfortunately, an extremely small proportion of individuals reported `capital_gain` and `capital_loss` in this study. These reflect real-world trends but are likely not representative of true income cateogires above and below $50,000.

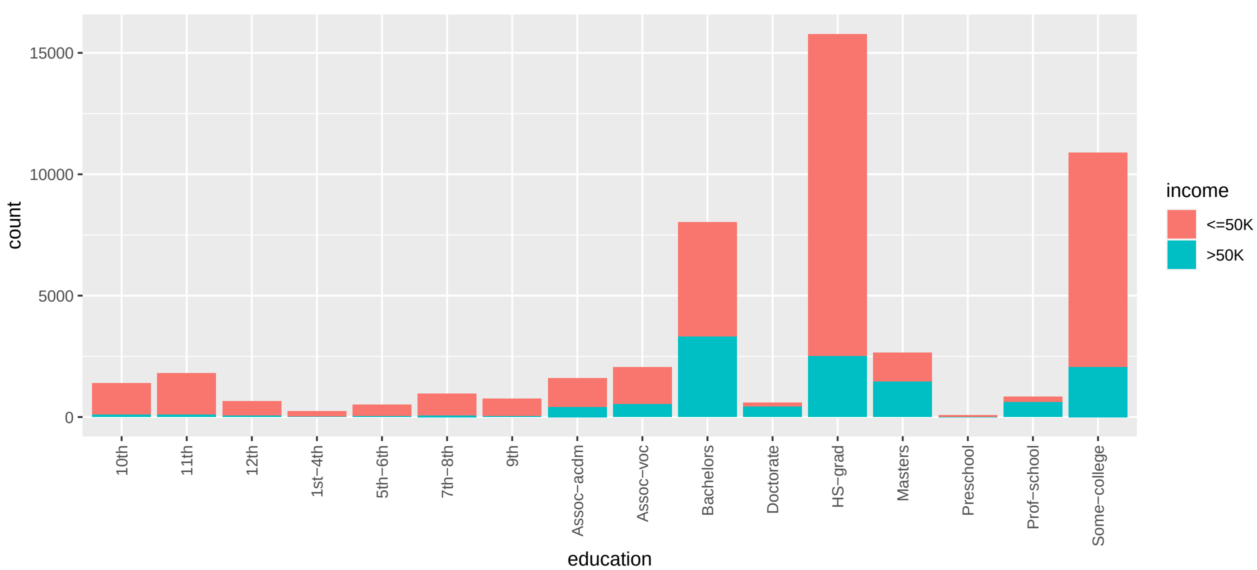
The next set of graphs shows the income distribution against `countries`, `workclass`, `education`, `sex` and `race`. We see male has higher income in both the categories than female. White race income distribution is significantly large as compared to other races. Private working classes earn more than any other categories and United States has largest income in both the categories compareds to all other countries. Another chart is shown below for details.




In addition to those trends we add that, many other countries with respondents have income less than or equal to $50,000. Of the other countries, only Canada and India appear to have a significant proportion of people with greater than $50,000 per year. The rest are either `?` or so small, they might as well be absent since they likely misrepresent the population of the country they are assigned to. There are no distributions similar to the scale, proportions, or magnitude of data captured for the United States which will effect our results.

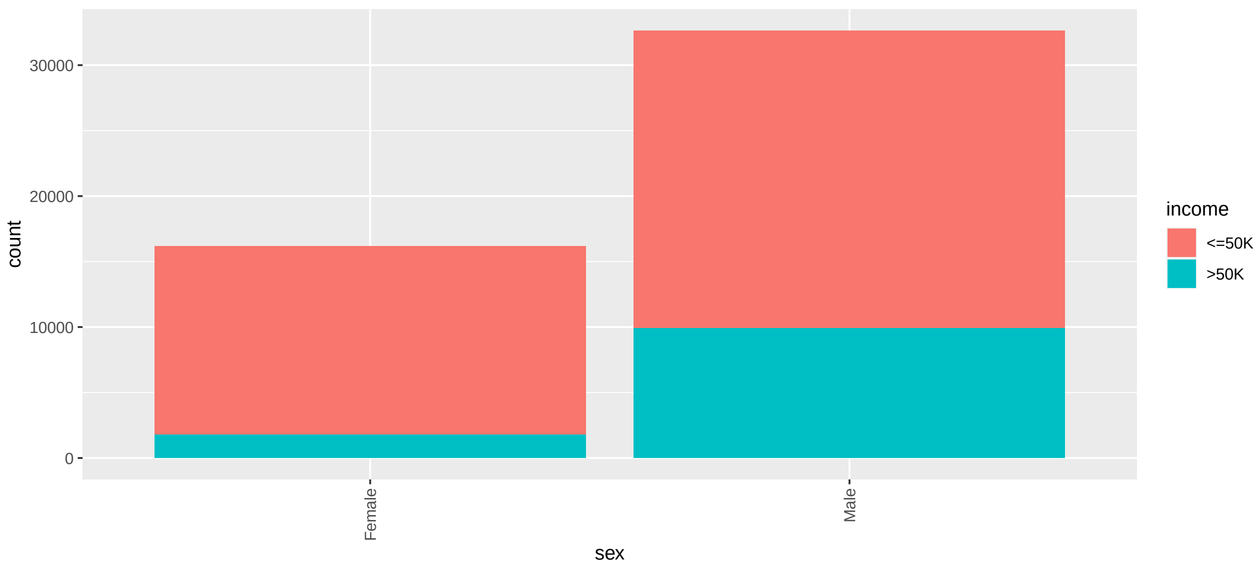



The private class dominates the proportion of respondents from this survey. If all other categories were stacked together, including those missing values labeled as `?` their values would barely cover half of the private working class individual's responses from this study. Because of this, it is also no surprise that the private working class holds the largest share of income in the category greater than $50,000. But what about education?

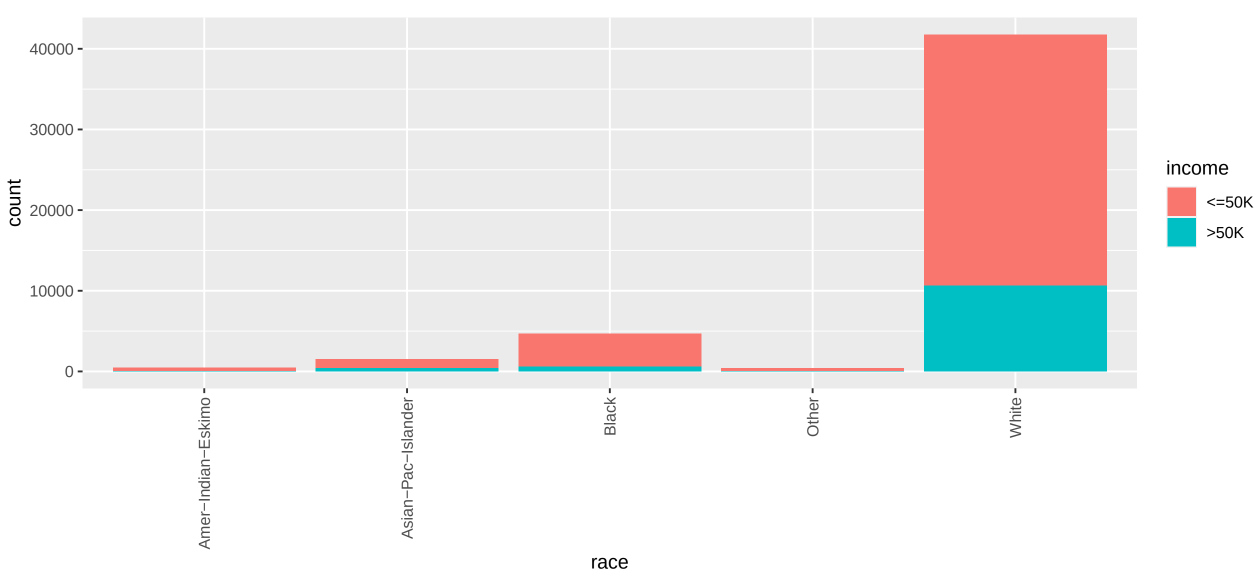



From this dataset, we see that education does have an impact of on earning potential. As years of education increase, the proportion of the population with income greater than $50,000 increases. However, the issue with this dataset is that is does not match repeated findings from peer-reviewed literature. Due to the slight increases in the proportion of people with `>50k`, it seems to matter less if you have a graduate or professional degree (categorized as `Masters`, `Doctorate`, `Prof-school` in the chart), than if you simply graduate from high school. We know this to be partially the case, but the income of the population of people with graduate or professional degrees should be greater in all cases since it also delays earning income (while costing the individual to pursue the degree).

This could be because we have a disproportionate amount of individuals who responded with at least some college when compared to the populations in real-world scenarios. This disproportionality is evident in the sum of the population of respondents with any education at or greater than the category `some_college`. Add up those with some_college, an associates, bachelors degree, masters, doctorate, and professional degrees and we have a population greater than or equal to all of high-school graduates. In other words, roughly 50 - 75% of this population would have been to college. Based on census bureau data, this is not the case in the real-world. Estimates in peer-reviewed literature places the proportion who have been to college well below our lowest bound. It is important to recognize this subset of the population which responded to the survey.



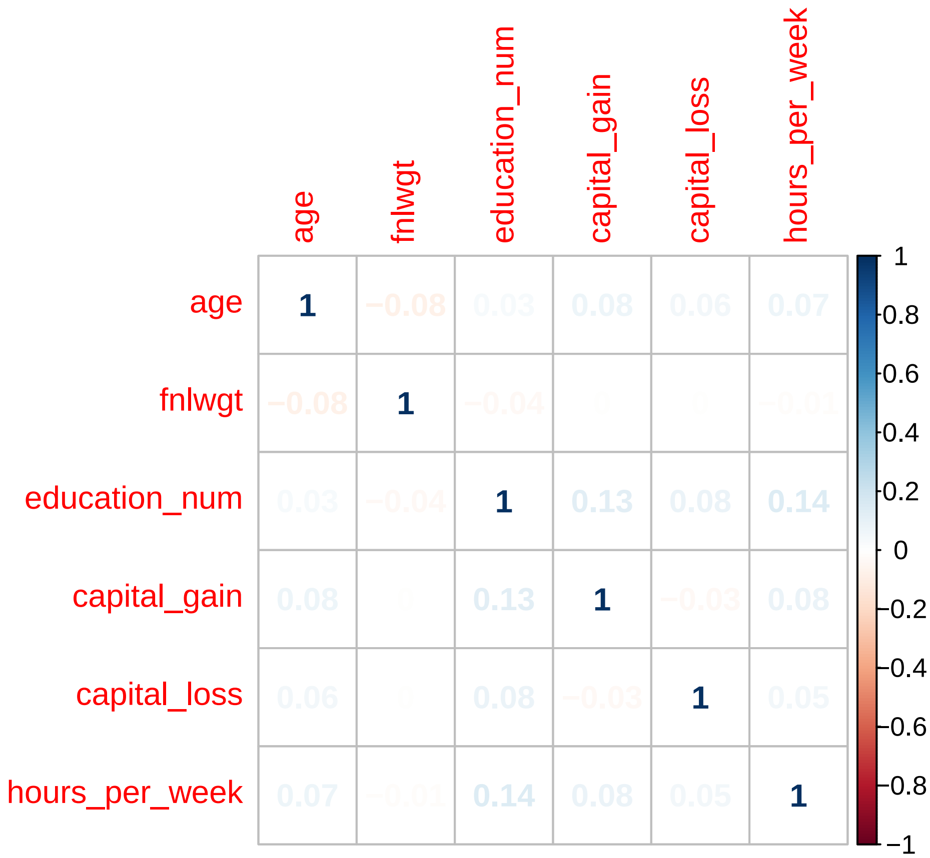

This chart shows how males hold most of the income amoung individuals in this dataset. Unfortunately, females makeup a smaller proportion of the dataset. This could be the case in some countries but in the U.S. (which makes up the greater than 90% of the data), it is common for women to have occupations that should pay the equivalent of men for the same occupation. However this chart shows that if the counts of each bar were adjusted to the same height, females would have a smaller number of individuals counted with income >50k based on this dataset. There is clearly not an equal income distribution among the sexes presented in this dataset.



The majority of individual respondents were white and as such, their distribution covers most of the chart. From this we can see that white individuals with `>50k` contains the largest of all the proportions of income among the races. Based on this dataset, it appears that the races with lowest proportion of people who have income greater than $50,000 are Black, American Indian Eskimos, or other minorities. This is consistent with conventional literature and it will also effect our results by making prediction of minority classes more difficult.

### Correlations

To determine how well each variable is correlated with our target variable and with one another, we construct a correlation plot. This plot contains the values of all correlation between variables represented by colors and numbers. The lighter the color, the lower the correlation. Meanwhile, darker blue indicates stronger positive correlations while darker red indicates stronger negative correlations.



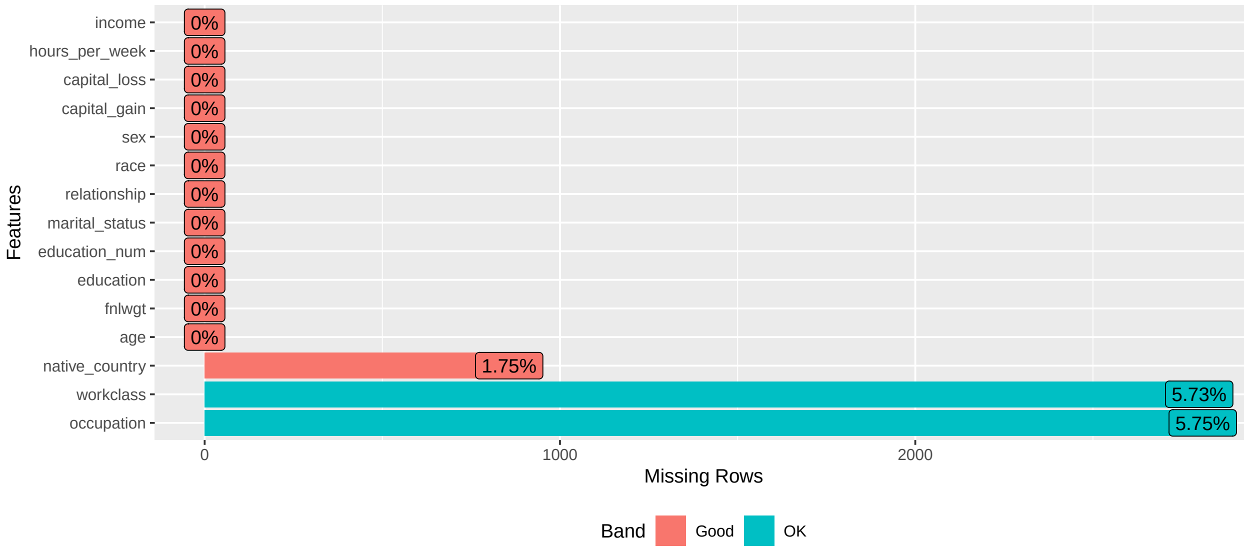

Given that our numeric features have correlation values near 0, they do not seem to be strongly correlated with our target. They also do not seem to have any correlation with one another so this is a factor that does not have to be dealt with. Weaker correlations indicated little to no interactions with our target variable.

## Data Preparation

Before this income data can be used as input in our machine learning models, it must be cleaned, formatted, and restructured — this is typically known as preprocessing. In this income dataset there are columns that have values listed as '?'. During the data preparation process we will clean these values, transform skewed features and perform train and test split for models. This preprocessing can help us with the outcome and significantly increase model accuracy of almost all our learning algorithms.

### Handling missing values

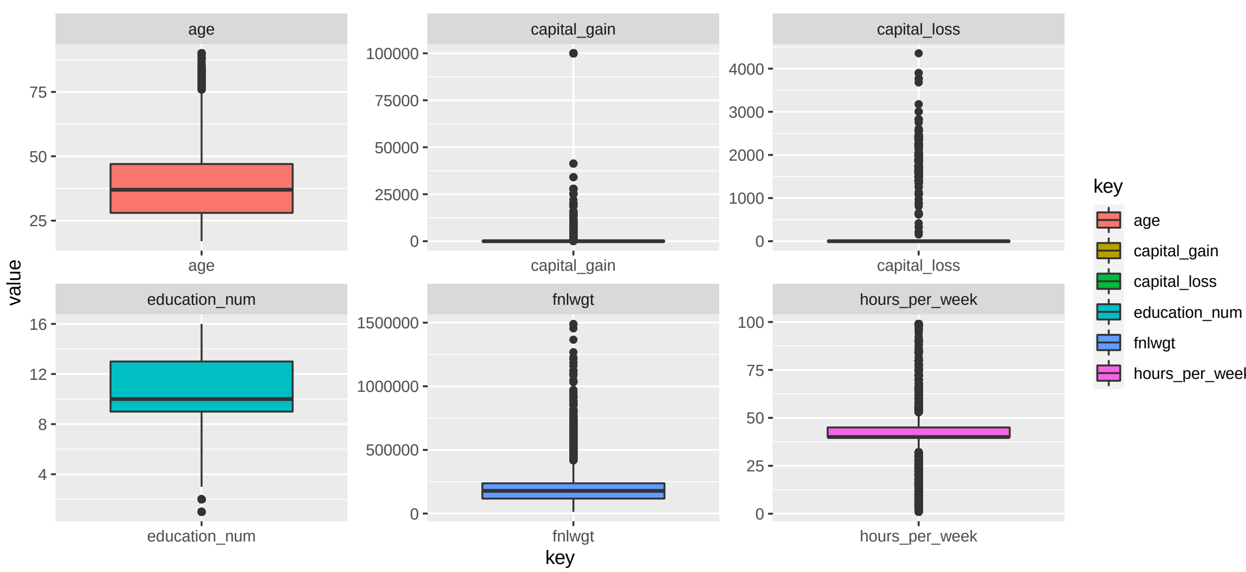
To this stage it is clear that our dataset does have missing values that appear as '?'. In the next step we replace the ? with NA and then take all the complete cases only. We do see there are 3620 cases with values missing and needs to be left out. We finally get the dataset with 45222 rows and 15 columns. A plot of the remaining proportions of missing values is shown after this reduction in dimesions of the dataset.




The only 3 variables that contained missing rows were `native_country`, `workclass`, and `occupation`. Each contains an acceptable amount of missing but these will not be highly weighted in our algorithms. We tally the number of incomplete cases present at this step. Show the reduction in dimensions of the dataset while maintaining the same number of variables.

```
## [1] 3620
```

```
## [1] 45222    15
```




There are many outliers identified in the `hours_per_week`, `capital_gain`, `capital_loss` and `fnlwgt` variables. These are shown as points in grid of boxplots above. The `fnlwgt` variable (i.e. final weight) should be removed since it has no predictive power and it is a feature to allocate similar weights to people with similar demographic characteristics. We are also removing variable `education` since it is just a label of `education_num` column. This will reduce our variables to 13 and our dimensions by 2.

## Preprocess using transformation

For highly-skewed feature distributions, we perform boxcox transformation for selected disproportionate columns to reduce the skewness and make it more Gaussian. Also combining the center and scale transforms standardizes the data. Now, the features will have a mean value of 0 and a standard deviation of 1. This preprocessing uses the caret package's 'preprocessing' function to return a box cox transformation on all numeric variables in our income dataset. These numeric variables include `age`, `education_num`, `capital_gain`, `capital_loss` and `hours_per_week`. A sample of the first few rows after this transformation are shown.

```
##           age        workclass education_num     marital_status
## 1  0.17111968        State-gov     1.1572034      Never-married
## 2  0.91061264 Self-emp-not-inc     1.1572034 Married-civ-spouse
## 3  0.09590960          Private    -0.4799097           Divorced
## 4  1.08942229          Private    -1.2243435 Married-civ-spouse
## 5 -0.75961330          Private     1.1572034 Married-civ-spouse
## 6  0.01909911          Private     1.5929027 Married-civ-spouse
##          occupation  relationship  race    sex capital_gain capital_loss
## 1      Adm-clerical Not-in-family White   Male    0.1428868   -0.2187778
## 2   Exec-managerial       Husband White   Male   -0.1467316   -0.2187778
## 3 Handlers-cleaners Not-in-family White   Male   -0.1467316   -0.2187778
## 4 Handlers-cleaners       Husband Black   Male   -0.1467316   -0.2187778
## 5    Prof-specialty          Wife Black Female   -0.1467316   -0.2187778
## 6   Exec-managerial          Wife White Female   -0.1467316   -0.2187778
##   hours_per_week native_country income
## 1     -0.0781192  United-States      0
## 2     -2.3267123  United-States      0
## 3     -0.0781192  United-States      0
## 4     -0.0781192  United-States      0
## 5     -0.0781192           Cuba      0
## 6     -0.0781192  United-States      0
```

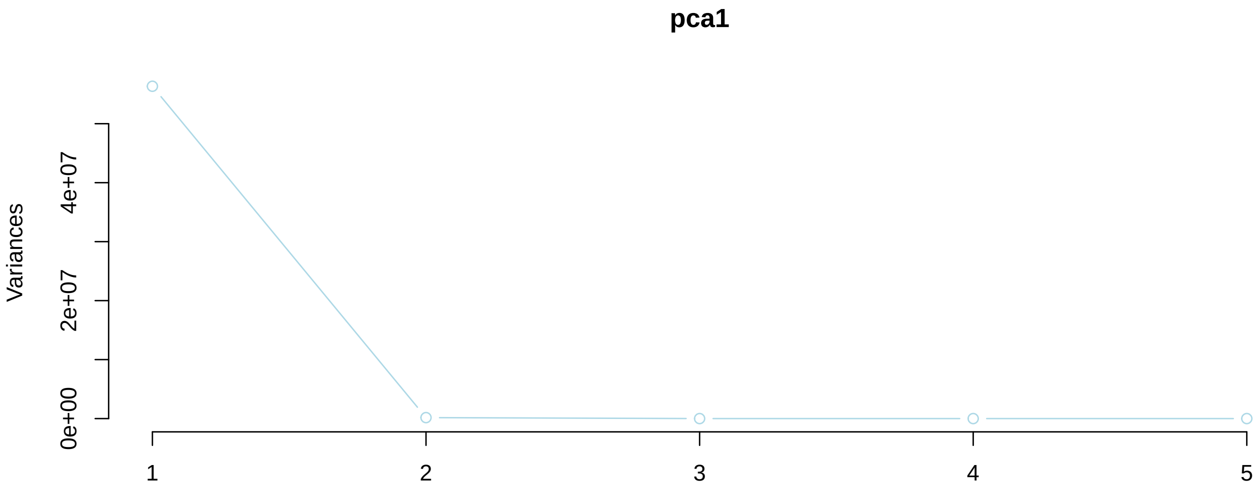
## PCA and Factor Analysis

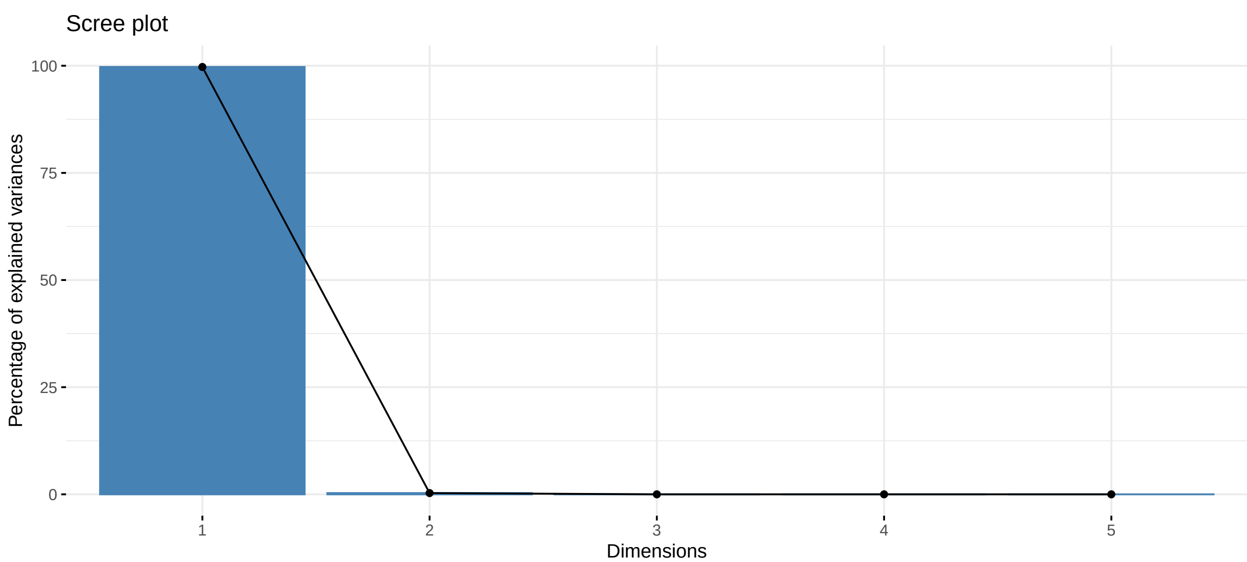
In this section we try to find those variables which could be used to reduce the dimensions of the dataset while also explaining the variation among the variables. Factors and numeric values typically require they own kinds of analysis, specifically factor analysis and principle component analysis or PCA. We take advantage of the multitude of levels identified within our factors during data exploration and assign numeric codes to each of them. This will allow all of our dataset's variables to appear numeric and be run in PCA while preserving the distance between points. We center and scale each run and complete 4 analyses. Ideally this will inform us of a few variables that we can use to maintain the same prediction accuracy in certain models. The results are shown below. Results are displayed below.

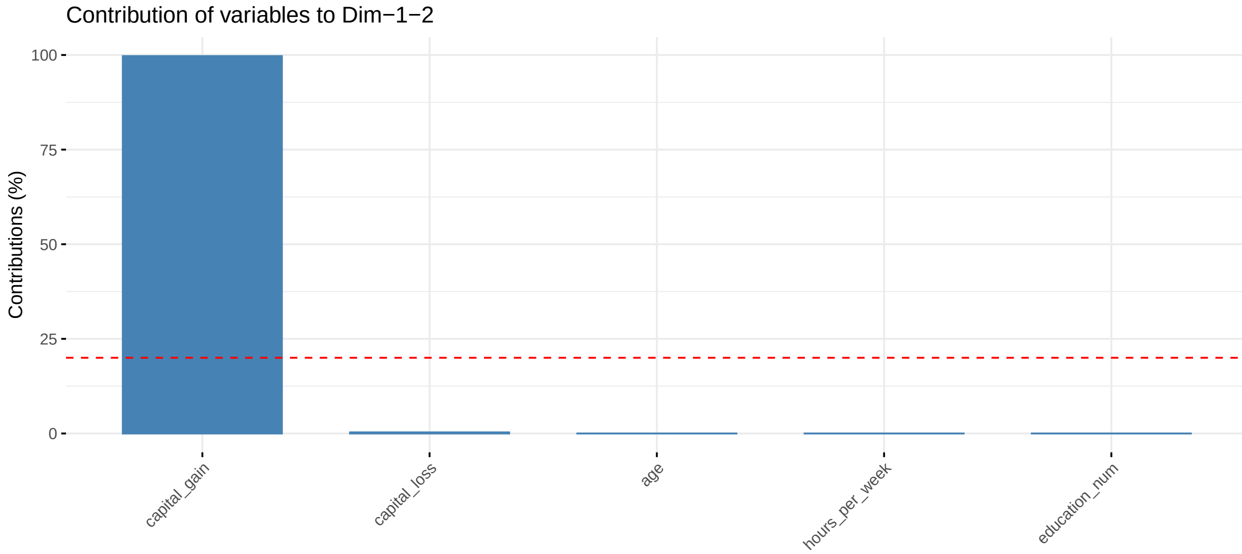
In the variance plots we notice how much in numerical terms the variation changes between PCA groups while the scree plots show the percentage of variance explained between those same groups. We also create contribution plots to show which variables contribute the most to the data, and bioplots to show their directional contribution (as high or low contribution ratings). These 4 methods should display a similar story but we make specific changes to notice the differences. Starting with all factor-type variables solely for PCA1 we run the algorithm. Then, we remove `capital_gain` and `capital_loss` variables. We repeat this on the full group of variables (numeric and factors) with and without these gain and loss variables since they skew the true results when trying to predict our target.

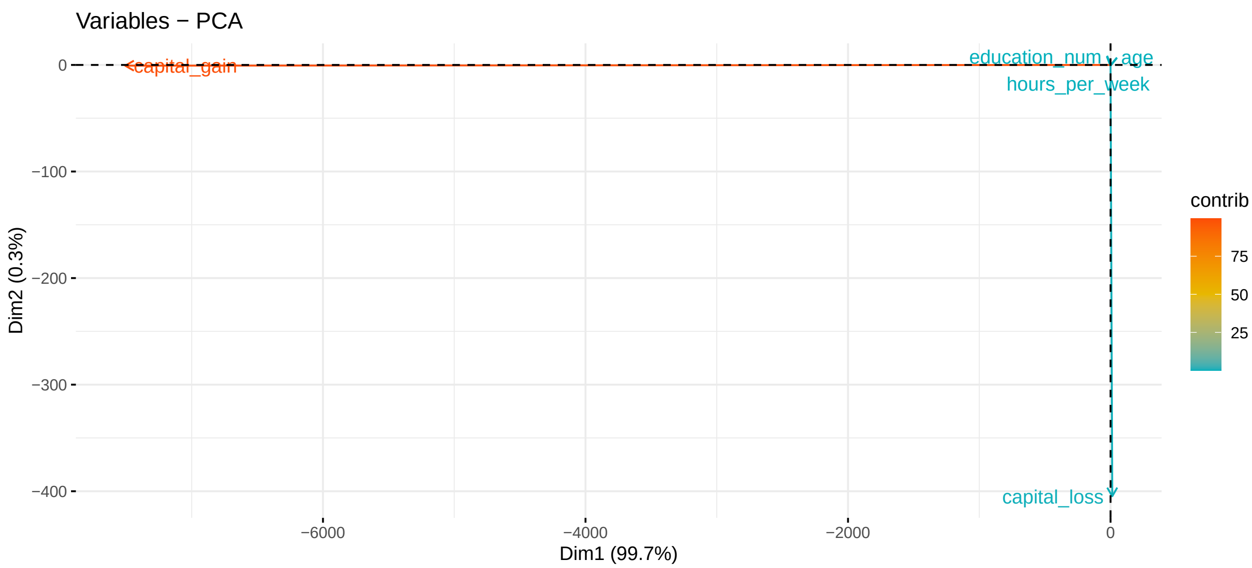
```
## Importance of components:
##                               PC1      PC2   PC3  PC4   PC5
```

```
## Standard deviation     7506.4415 404.7482 13.37 11.7 2.499
## Proportion of Variance    0.9971   0.0029  0.00  0.0 0.000
## Cumulative Proportion     0.9971   1.0000  1.00  1.0 1.000
```

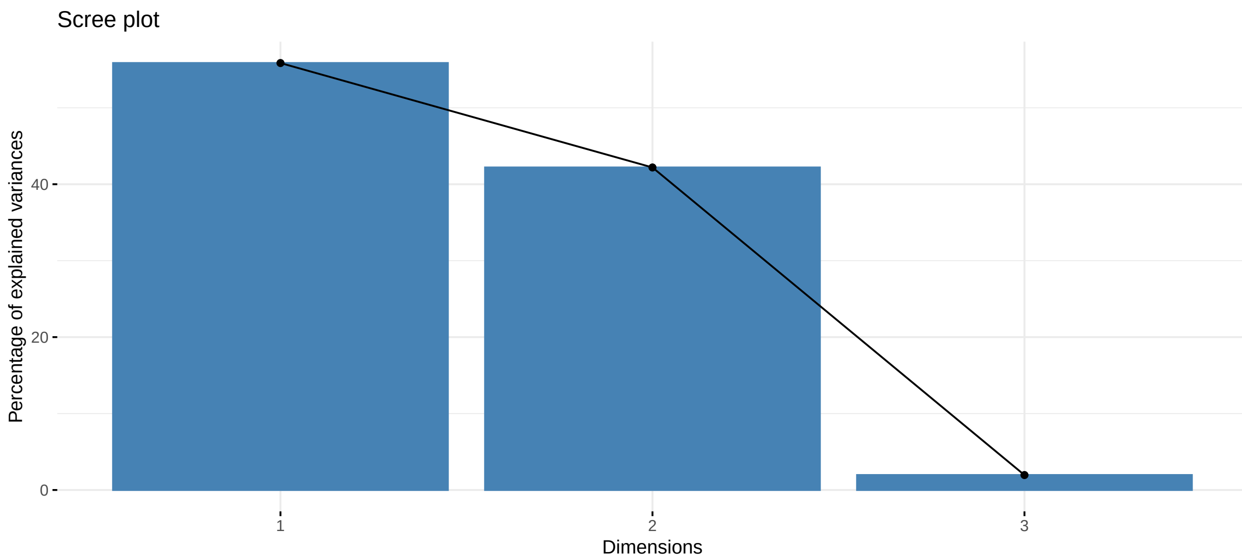


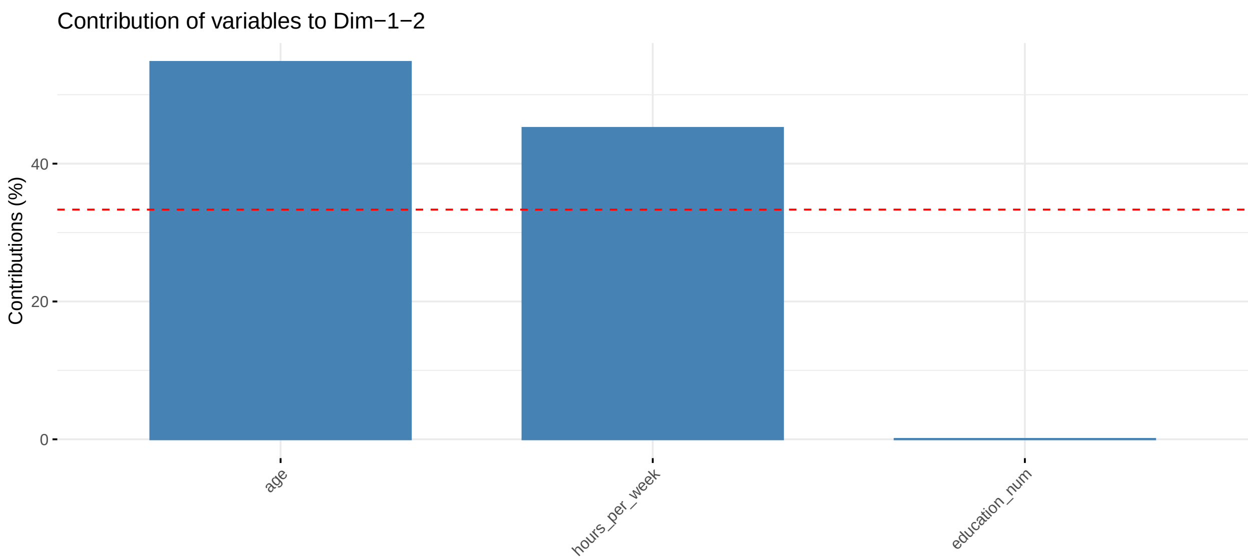



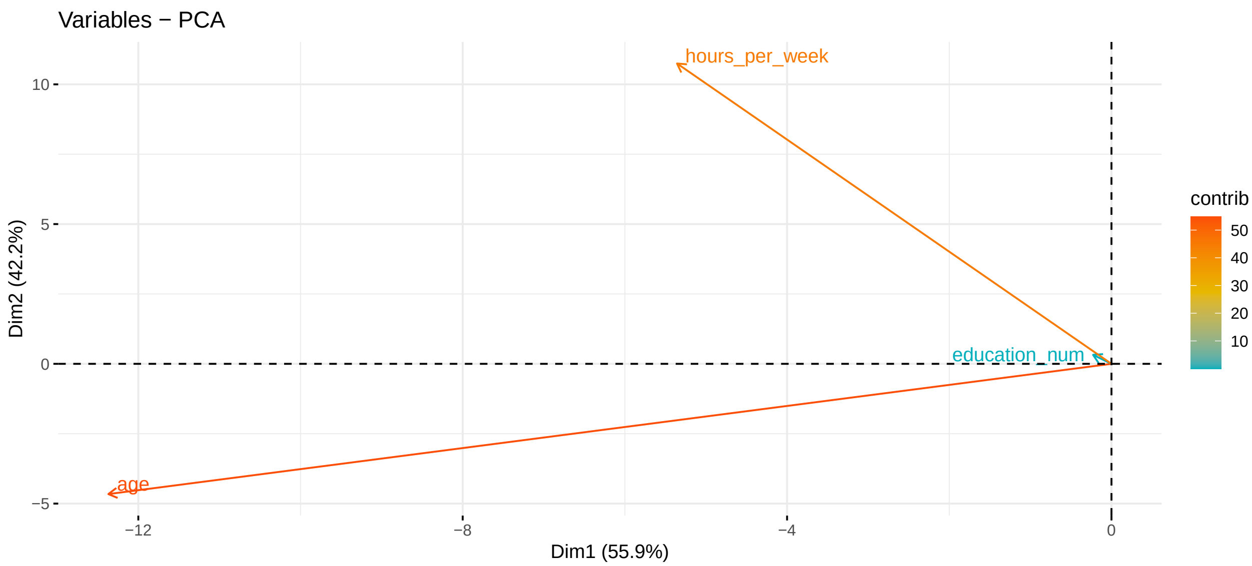


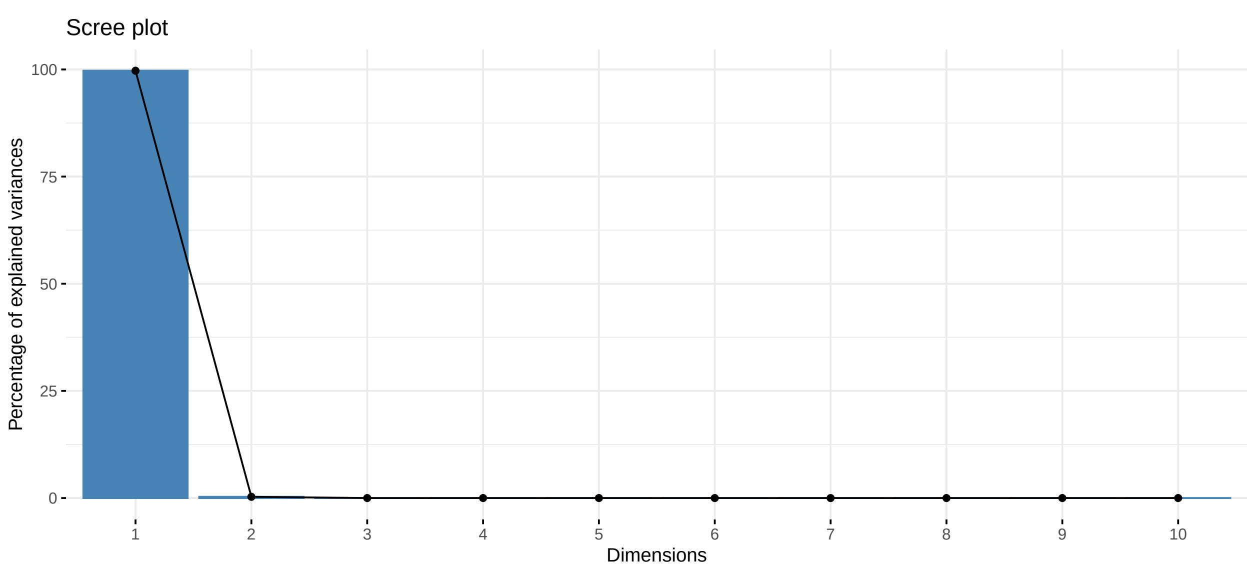

```
## Importance of components:
##                            PC1     PC2     PC3
## Standard deviation     13.4813 11.7173 2.52352
## Proportion of Variance  0.5585  0.4219 0.01957
## Cumulative Proportion   0.5585  0.9804 1.00000
```

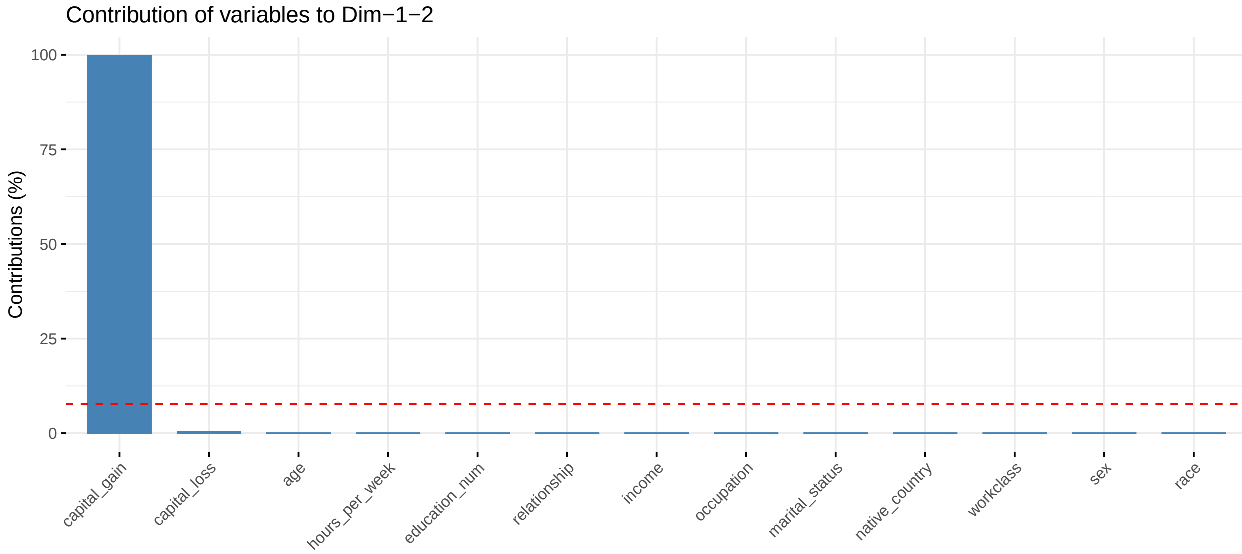

Scree plot
Percentage of explained variances
0
20
40
1
2
3
Dimensions

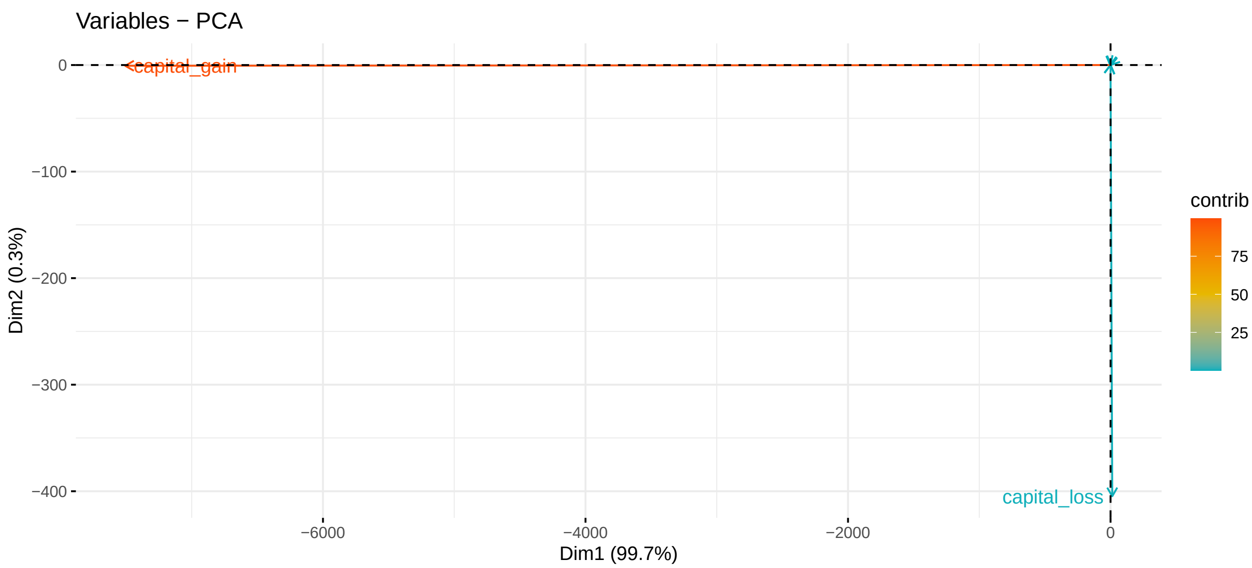

Contribution of variables to Dim−1−2
Contributions (%)
0
20
40
age
hours_per_week
education_num



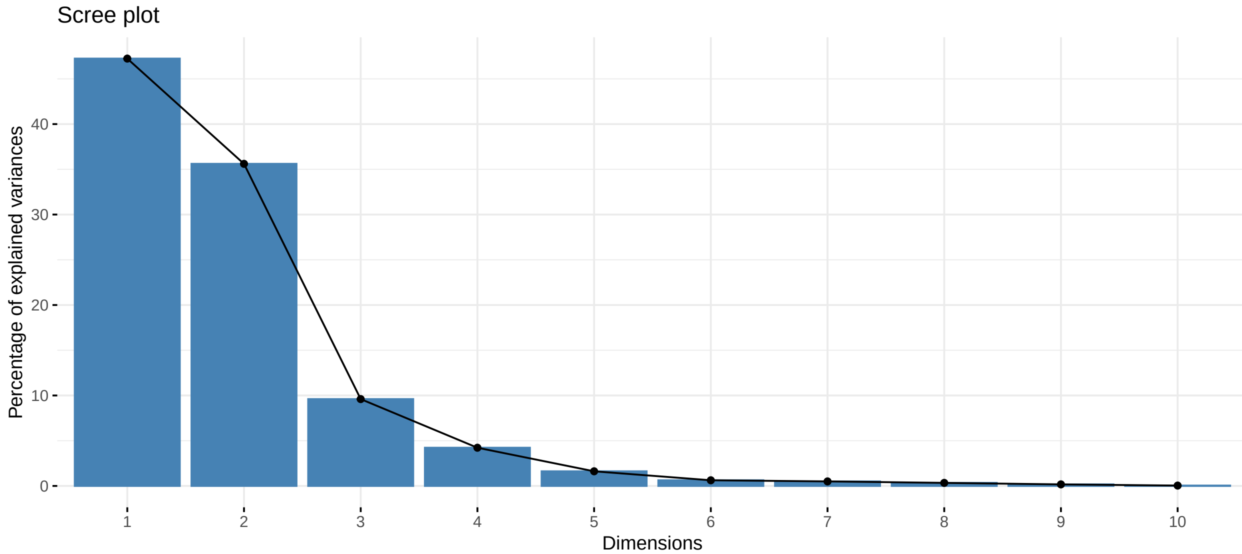

```
## Importance of components:
##                              PC1      PC2   PC3  PC4   PC5   PC6   PC7   PC8
## Standard deviation     7506.4415 404.7483 13.39 11.7 6.086 4.035 2.476 1.553
## Proportion of Variance    0.9971   0.0029  0.00  0.0 0.000 0.000 0.000 0.000
## Cumulative Proportion     0.9971   1.0000  1.00  1.0 1.000 1.000 1.000 1.000
##                          PC9  PC10   PC11   PC12   PC13
## Standard deviation     1.391 1.141 0.8144 0.3902 0.3513
## Proportion of Variance 0.000 0.000 0.0000 0.0000 0.0000
## Cumulative Proportion  1.000 1.000 1.0000 1.0000 1.0000
```

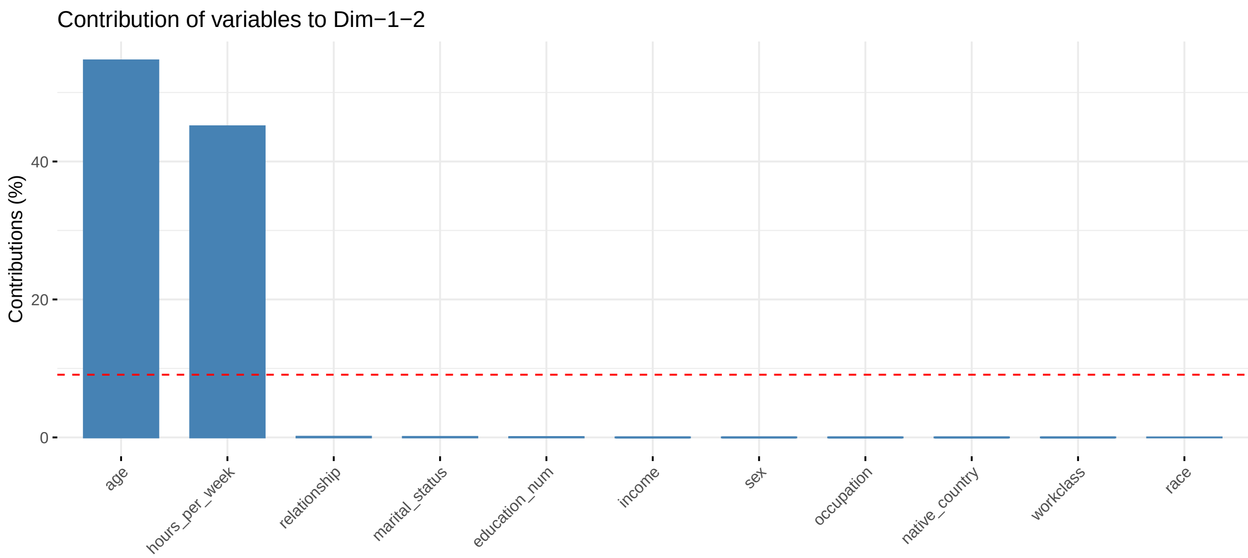


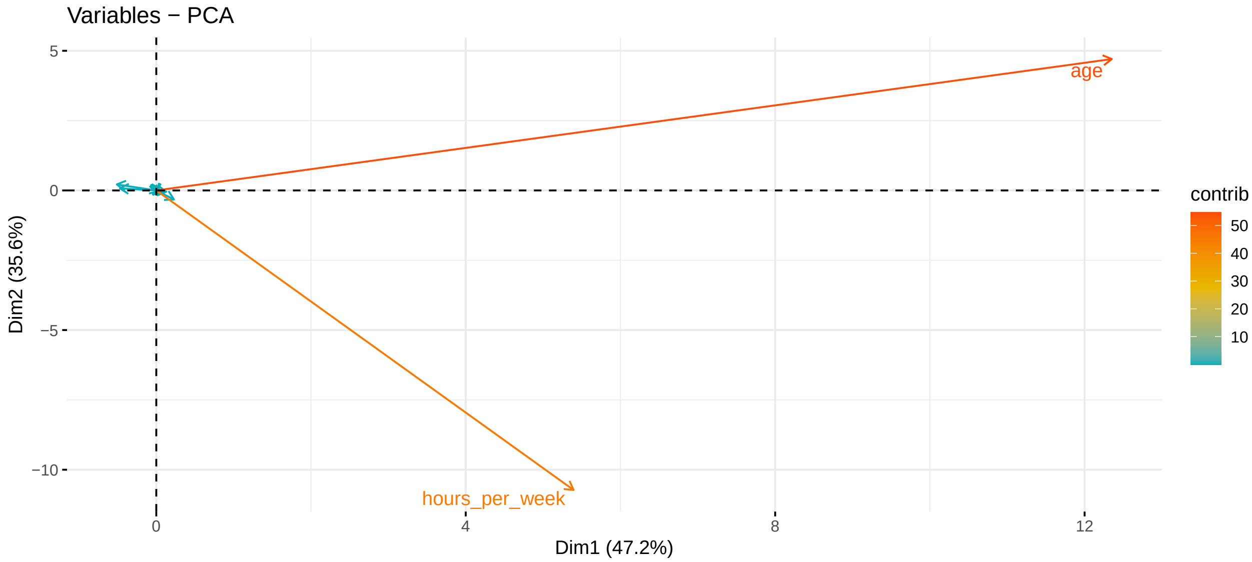



```
## Importance of components:
##                            PC1     PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     13.4999 11.7204 6.08599 4.03695 2.49933 1.55425 1.39069
## Proportion of Variance  0.4724  0.3561 0.09601 0.04224 0.01619 0.00626 0.00501
## Cumulative Proportion   0.4724  0.8285 0.92446 0.96671 0.98290 0.98916 0.99417
##                            PC8     PC9    PC10    PC11
## Standard deviation     1.14165 0.81443 0.39604 0.35384
## Proportion of Variance 0.00338 0.00172 0.00041 0.00032
## Cumulative Proportion  0.99755 0.99927 0.99968 1.00000
```


Scree plot
Percentage of explained variances
Dimensions
0
10
20
30
40
1
2
3
4
5
6
7
8
9
10


Contribution of variables to Dim−1−2
Contributions (%)
0
20
40
age
hours_per_week
relationship
marital_status
education_num
income
sex
occupation
native_country
workclass
race




When we exclude the `capital_gain` and `capital_loss` variables, we find that the most important factors for explaining the variation in the data are the age of the individual, how many hours per week they work, and what their education level is. These factors are largely consistent with real-world expectations, even though we are aware the dataset is not necessarily representative of a larger population. These variables: `age`, `education_num`, and `hours_per_week` should be used if a reduction in the dimensions of the data is necessary for model development while maintaining accuracy.

### Training and Test Partition

In this step for data preparation we will partition the training dataset into training and validation sets using the `createDataPartition` method from the `caret` package. We will reserve 70% for training and rest 30% for validation purpose. The dimensions of our training dataset become 31656 observations of our 13 selected variables. We place the remaining 13566 observations of individuals aside to assess our models performances.

## Build Models

With data prepared for modeling, we develop several models that we suspect would have the best chances of improving our prediction of the binary income target. This includes logisitc regression, decision trees, a random forest model, clustering techniques and more. We use the same data set for each without further transformations or reversion to keep the results simplistic. We begin in this section, tabulating the accuracy of each model to build additional models that add to the strengths and cover the weakness (and error prone) portions of certain models. Those results are then compiled in the model performance section.

### Logistic Regression

Our first model is a logistic regression model. This model will let us identify variables and factor levels that have significant influence over the target variable income. To do this we must first convert factors into dummy variables within the training and test sets. We utilize the caret package for its development. Results are shown.

```
##
## Call:
## glm(formula = income ~ ., family = "binomial", data = training)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
```

```
## -4.8857  -0.5154  -0.1875  -0.0215   3.9965
##
## Coefficients:
##                                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)                             -3.47311    0.74384  -4.669 3.02e-06
## age                                      0.42398    0.02391  17.735  < 2e-16
## workclassLocal-gov                      -0.70800    0.10982  -6.447 1.14e-10
## workclassPrivate                        -0.48537    0.09074  -5.349 8.84e-08
## workclassSelf-emp-inc                   -0.35680    0.11986  -2.977 0.002912
## workclassSelf-emp-not-inc               -1.05975    0.10651  -9.950  < 2e-16
## workclassState-gov                      -0.79442    0.12079  -6.577 4.81e-11
## workclassWithout-pay                    -1.17435    0.82319  -1.427 0.153701
## education_num                            0.70257    0.02325  30.219  < 2e-16
## marital_statusMarried-AF-spouse          2.39636    0.61699   3.884 0.000103
## marital_statusMarried-civ-spouse         2.42026    0.27689   8.741  < 2e-16
## marital_statusMarried-spouse-absent      0.22617    0.22609   1.000 0.317132
## marital_statusNever-married             -0.35877    0.08800  -4.077 4.56e-05
## marital_statusSeparated                  0.12102    0.15507   0.780 0.435156
## marital_statusWidowed                    0.16393    0.15205   1.078 0.280983
## occupationArmed-Forces                   0.34511    0.93830   0.368 0.713024
## occupationCraft-repair                   0.01538    0.07826   0.196 0.844244
## occupationExec-managerial                0.71541    0.07549   9.477  < 2e-16
## occupationFarming-fishing               -0.99588    0.13732  -7.253 4.09e-13
## occupationHandlers-cleaners             -0.66232    0.13622  -4.862 1.16e-06
## occupationMachine-op-inspct             -0.37912    0.10149  -3.735 0.000187
## occupationOther-service                 -0.87291    0.11601  -7.525 5.28e-14
## occupationPriv-house-serv               -1.76899    0.77207  -2.291 0.021951
## occupationProf-specialty                 0.52309    0.07837   6.675 2.47e-11
## occupationProtective-serv                0.49507    0.12565   3.940 8.15e-05
## occupationSales                          0.26105    0.08013   3.258 0.001123
## occupationTech-support                   0.52252    0.10809   4.834 1.34e-06
## occupationTransport-moving              -0.10875    0.09657  -1.126 0.260104
## relationshipNot-in-family                0.65806    0.27368   2.404 0.016196
## relationshipOther-relative              -0.57552    0.25459  -2.261 0.023785
## relationshipOwn-child                   -0.35996    0.26685  -1.349 0.177358
## relationshipUnmarried                    0.43244    0.28956   1.493 0.135321
## relationshipWife                         1.25354    0.10318  12.149  < 2e-16
## raceAsian-Pac-Islander                   1.08237    0.27466   3.941 8.12e-05
## raceBlack                                0.47047    0.23270   2.022 0.043201
## raceOther                                0.75004    0.34122   2.198 0.027943
## raceWhite                                0.70143    0.22136   3.169 0.001531
## sexMale                                  0.81226    0.07848  10.349  < 2e-16
## capital_gain                             2.40587    0.07870  30.569  < 2e-16
## capital_loss                             0.26919    0.01523  17.674  < 2e-16
## hours_per_week                           0.33501    0.01947  17.206  < 2e-16
## native_countryCanada                    -0.27713    0.68660  -0.404 0.686493
## native_countryChina                     -1.46388    0.70671  -2.071 0.038321
## native_countryColumbia                  -2.58076    1.02850  -2.509 0.012099
## native_countryCuba                      -0.59564    0.70644  -0.843 0.399135
## native_countryDominican-Republic        -2.65882    1.21353  -2.191 0.028453
## native_countryEcuador                   -1.51333    0.99685  -1.518 0.128988
## native_countryEl-Salvador               -0.95062    0.79168  -1.201 0.229846
## native_countryEngland                   -0.21348    0.72118  -0.296 0.767215
## native_countryFrance                     0.21584    0.80953   0.267 0.789754
```

```
## native_countryGermany                    -0.55407    0.68625  -0.807 0.419442
## native_countryGreece                     -0.72748    0.76363  -0.953 0.340763
## native_countryGuatemala                  -0.92641    0.99291  -0.933 0.350807
## native_countryHaiti                       0.29994    0.82510   0.364 0.716217
## native_countryHoland-Netherlands         -6.79532  119.47006  -0.057 0.954642
## native_countryHonduras                   -0.28313    1.32242  -0.214 0.830470
## native_countryHong                       -1.09882    0.86073  -1.277 0.201737
## native_countryHungary                     0.04357    0.88496   0.049 0.960733
## native_countryIndia                      -1.22863    0.68149  -1.803 0.071411
## native_countryIran                       -0.25848    0.78130  -0.331 0.740769
## native_countryIreland                     0.51969    0.85381   0.609 0.542741
## native_countryItaly                       0.30977    0.70855   0.437 0.661976
## native_countryJamaica                    -0.17515    0.81132  -0.216 0.829076
## native_countryJapan                      -0.51107    0.72863  -0.701 0.483051
## native_countryLaos                       -1.94923    1.28405  -1.518 0.129006
## native_countryMexico                     -1.05230    0.66693  -1.578 0.114604
## native_countryNicaragua                  -1.02472    1.00870  -1.016 0.309684
## native_countryOutlying-US(Guam-USVI-etc) -0.89973    1.25993  -0.714 0.475161
## native_countryPeru                       -2.11946    1.04829  -2.022 0.043195
## native_countryPhilippines                -0.72138    0.65812  -1.096 0.273025
## native_countryPoland                     -0.21904    0.73221  -0.299 0.764830
## native_countryPortugal                   -0.19556    0.79784  -0.245 0.806371
## native_countryPuerto-Rico                -0.77693    0.72086  -1.078 0.281130
## native_countryScotland                   -2.62038    1.26438  -2.072 0.038222
## native_countrySouth                      -2.39725    0.76145  -3.148 0.001642
## native_countryTaiwan                     -0.77156    0.77061  -1.001 0.316712
## native_countryThailand                   -1.97936    1.05882  -1.869 0.061568
## native_countryTrinadad&Tobago            -1.28077    1.05847  -1.210 0.226273
## native_countryUnited-States              -0.40577    0.63489  -0.639 0.522746
## native_countryVietnam                    -1.92562    0.86961  -2.214 0.026805
## native_countryYugoslavia                  1.17984    1.02311   1.153 0.248832
##
## (Intercept)                              ***
## age                                      ***
## workclassLocal-gov                       ***
## workclassPrivate                         ***
## workclassSelf-emp-inc                    **
## workclassSelf-emp-not-inc                ***
## workclassState-gov                       ***
## workclassWithout-pay
## education_num                            ***
## marital_statusMarried-AF-spouse          ***
## marital_statusMarried-civ-spouse         ***
## marital_statusMarried-spouse-absent
## marital_statusNever-married              ***
## marital_statusSeparated
## marital_statusWidowed
## occupationArmed-Forces
## occupationCraft-repair
## occupationExec-managerial                ***
## occupationFarming-fishing                ***
## occupationHandlers-cleaners              ***
## occupationMachine-op-inspct              ***
## occupationOther-service                  ***
```

```
## occupationPriv-house-serv             *
## occupationProf-specialty              ***
## occupationProtective-serv             ***
## occupationSales                       **
## occupationTech-support                ***
## occupationTransport-moving
## relationshipNot-in-family             *
## relationshipOther-relative            *
## relationshipOwn-child
## relationshipUnmarried
## relationshipWife                      ***
## raceAsian-Pac-Islander                ***
## raceBlack                             *
## raceOther                             *
## raceWhite                             **
## sexMale                               ***
## capital_gain                          ***
## capital_loss                          ***
## hours_per_week                        ***
## native_countryCanada
## native_countryChina                   *
## native_countryColumbia                *
## native_countryCuba
## native_countryDominican-Republic      *
## native_countryEcuador
## native_countryEl-Salvador
## native_countryEngland
## native_countryFrance
## native_countryGermany
## native_countryGreece
## native_countryGuatemala
## native_countryHaiti
## native_countryHoland-Netherlands
## native_countryHonduras
## native_countryHong
## native_countryHungary
## native_countryIndia                   .
## native_countryIran
## native_countryIreland
## native_countryItaly
## native_countryJamaica
## native_countryJapan
## native_countryLaos
## native_countryMexico
## native_countryNicaragua
## native_countryOutlying-US(Guam-USVI-etc)
## native_countryPeru                    *
## native_countryPhilippines
## native_countryPoland
## native_countryPortugal
## native_countryPuerto-Rico
## native_countryScotland                *
## native_countrySouth                   **
## native_countryTaiwan
```

```
## native_countryThailand                   .
## native_countryTrinadad&Tobago
## native_countryUnited-States
## native_countryVietnam                    *
## native_countryYugoslavia
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 35452  on 31655  degrees of freedom
## Residual deviance: 20411  on 31575  degrees of freedom
## AIC: 20573
##
## Number of Fisher Scoring iterations: 9
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 9449  755
##          1 1307 2055
##
##                Accuracy : 0.848
##                  95% CI : (0.8418, 0.854)
##     No Information Rate : 0.7929
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5685
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.8785
##             Specificity : 0.7313
##          Pos Pred Value : 0.9260
##          Neg Pred Value : 0.6112
##              Prevalence : 0.7929
##          Detection Rate : 0.6965
##    Detection Prevalence : 0.7522
##       Balanced Accuracy : 0.8049
##
##        'Positive' Class : 0
##
```

Our logistic regression model accuracy comes out as 0.848 or roughly 85%. There is room for improvement in this model's sensitivity among other variables. By observation of the significance for each variables coefficient of the first logistic regression model, the county columns do no provide much, if any, significance. Next, we'll try to improve the overall performance of the model to remove the country as a variable.

```
##
## Call:
## glm(formula = income ~ age + workclass + education_num + marital_status +
##     occupation + relationship + race + sex + capital_gain + capital_loss +
##     hours_per_week, family = "binomial", data = training)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.1765  -0.5190  -0.1897  -0.0248   3.8727
##
## Coefficients:
##                                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)                          -3.84375    0.37085 -10.365  < 2e-16 ***
## age                                   0.42704    0.02377  17.964  < 2e-16 ***
## workclassLocal-gov                   -0.70705    0.10957  -6.453 1.10e-10 ***
## workclassPrivate                     -0.49229    0.09054  -5.437 5.42e-08 ***
## workclassSelf-emp-inc                -0.36874    0.11938  -3.089  0.00201 **
## workclassSelf-emp-not-inc            -1.07250    0.10619 -10.100  < 2e-16 ***
## workclassState-gov                   -0.79835    0.12063  -6.618 3.63e-11 ***
## workclassWithout-pay                 -1.15359    0.82450  -1.399  0.16177
## education_num                         0.70247    0.02289  30.693  < 2e-16 ***
## marital_statusMarried-AF-spouse       2.38860    0.61396   3.891  0.00010 ***
## marital_statusMarried-civ-spouse      2.38313    0.27427   8.689  < 2e-16 ***
## marital_statusMarried-spouse-absent  0.20679    0.22395   0.923  0.35581
## marital_statusNever-married          -0.35864    0.08777  -4.086 4.39e-05 ***
## marital_statusSeparated               0.11248    0.15483   0.726  0.46757
## marital_statusWidowed                 0.15843    0.15196   1.043  0.29712
## occupationArmed-Forces                0.35864    0.93893   0.382  0.70248
## occupationCraft-repair                0.01981    0.07798   0.254  0.79945
## occupationExec-managerial             0.72302    0.07521   9.613  < 2e-16 ***
## occupationFarming-fishing            -0.98003    0.13654  -7.178 7.09e-13 ***
## occupationHandlers-cleaners          -0.66306    0.13574  -4.885 1.03e-06 ***
## occupationMachine-op-inspct          -0.39751    0.10093  -3.939 8.19e-05 ***
## occupationOther-service              -0.88180    0.11518  -7.656 1.92e-14 ***
## occupationPriv-house-serv            -1.77814    0.76588  -2.322  0.02025 *
## occupationProf-specialty              0.52934    0.07805   6.782 1.19e-11 ***
## occupationProtective-serv             0.48854    0.12510   3.905 9.41e-05 ***
## occupationSales                       0.26084    0.07988   3.265  0.00109 **
## occupationTech-support                0.53429    0.10783   4.955 7.24e-07 ***
## occupationTransport-moving           -0.09862    0.09621  -1.025  0.30536
## relationshipNot-in-family             0.63300    0.27104   2.335  0.01952 *
## relationshipOther-relative           -0.63392    0.25152  -2.520  0.01172 *
## relationshipOwn-child                -0.37860    0.26498  -1.429  0.15307
## relationshipUnmarried                 0.39173    0.28695   1.365  0.17220
## relationshipWife                      1.24731    0.10285  12.128  < 2e-16 ***
## raceAsian-Pac-Islander                0.57791    0.24371   2.371  0.01773 *
## raceBlack                             0.48339    0.23209   2.083  0.03727 *
## raceOther                             0.45988    0.33398   1.377  0.16852
## raceWhite                             0.70150    0.22118   3.172  0.00152 **
## sexMale                               0.80531    0.07829  10.286  < 2e-16 ***
## capital_gain                          2.40571    0.07844  30.671  < 2e-16 ***
## capital_loss                          0.26819    0.01516  17.694  < 2e-16 ***
## hours_per_week                        0.33279    0.01938  17.176  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 35452  on 31655  degrees of freedom
## Residual deviance: 20515  on 31615  degrees of freedom
```

```
## AIC: 20597
##
## Number of Fisher Scoring iterations: 7
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 9449  755
##          1 1307 2055
##
##                Accuracy : 0.848
##                  95% CI : (0.8418, 0.854)
##     No Information Rate : 0.7929
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5685
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.8785
##             Specificity : 0.7313
##          Pos Pred Value : 0.9260
##          Neg Pred Value : 0.6112
##              Prevalence : 0.7929
##          Detection Rate : 0.6965
##    Detection Prevalence : 0.7522
##       Balanced Accuracy : 0.8049
##
##        'Positive' Class : 0
##
```

Our logistic regression without countries model accuracy comes out as 0.848 or roughly 85%. There is room for improvement in this model's sensitivity among other variables. We try to improve this by adding the country variable back and creating dummy variables for the variables that are factors.

```
##
## Call:
## glm(formula = income ~ ., family = "binomial", data = training.dum)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.8857  -0.5154  -0.1875  -0.0215   3.9965
##
## Coefficients: (11 not defined because of singularities)
##                                               Estimate Std. Error z value
## (Intercept)                                   -0.64521    1.19892  -0.538
## age                                            0.42398    0.02391  17.735
## education_num                                  0.70257    0.02325  30.219
## capital_gain                                   2.40587    0.07870  30.569
## capital_loss                                   0.26919    0.01523  17.674
## hours_per_week                                 0.33501    0.01947  17.206
## `workclass_?`                                       NA         NA      NA
## `workclass_Federal-gov`                        1.17435    0.82319   1.427
## `workclass_Local-gov`                          0.46635    0.82148   0.568
```

```
## `workclass_Never-worked`                           NA         NA      NA
## workclass_Private                             0.68898    0.81882   0.841
## `workclass_Self-emp-inc`                      0.81755    0.82229   0.994
## `workclass_Self-emp-not-inc`                  0.11461    0.82022   0.140
## `workclass_State-gov`                         0.37993    0.82304   0.462
## `workclass_Without-pay`                            NA         NA      NA
## marital_status_Divorced                      -0.16393    0.15205  -1.078
## `marital_status_Married-AF-spouse`            2.23243    0.63132   3.536
## `marital_status_Married-civ-spouse`           2.25633    0.30620   7.369
## `marital_status_Married-spouse-absent`        0.06224    0.25964   0.240
## `marital_status_Never-married`               -0.52270    0.15788  -3.311
## marital_status_Separated                     -0.04292    0.20055  -0.214
## marital_status_Widowed                             NA         NA      NA
## `occupation_?`                                     NA         NA      NA
## `occupation_Adm-clerical`                     0.10875    0.09657   1.126
## `occupation_Armed-Forces`                     0.45385    0.94018   0.483
## `occupation_Craft-repair`                     0.12413    0.08290   1.497
## `occupation_Exec-managerial`                  0.82416    0.08511   9.683
## `occupation_Farming-fishing`                 -0.88713    0.13867  -6.397
## `occupation_Handlers-cleaners`               -0.55357    0.13953  -3.967
## `occupation_Machine-op-inspct`               -0.27037    0.10549  -2.563
## `occupation_Other-service`                   -0.76416    0.12250  -6.238
## `occupation_Priv-house-serv`                 -1.66024    0.77356  -2.146
## `occupation_Prof-specialty`                   0.63184    0.09146   6.908
## `occupation_Protective-serv`                  0.60382    0.13031   4.634
## occupation_Sales                              0.36980    0.08748   4.227
## `occupation_Tech-support`                     0.63127    0.11616   5.434
## `occupation_Transport-moving`                      NA         NA      NA
## relationship_Husband                         -1.25354    0.10318 -12.149
## `relationship_Not-in-family`                 -0.59548    0.28428  -2.095
## `relationship_Other-relative`                -1.82907    0.26480  -6.907
## `relationship_Own-child`                     -1.61350    0.27749  -5.815
## relationship_Unmarried                       -0.82110    0.29384  -2.794
## relationship_Wife                                  NA         NA      NA
## `race_Amer-Indian-Eskimo`                    -0.70143    0.22136  -3.169
## `race_Asian-Pac-Islander`                     0.38095    0.16545   2.302
## race_Black                                   -0.23096    0.07771  -2.972
## race_Other                                    0.04861    0.26111   0.186
## race_White                                         NA         NA      NA
## sex_Female                                   -0.81226    0.07848 -10.349
## sex_Male                                           NA         NA      NA
## `native_country_?`                                 NA         NA      NA
## native_country_Cambodia                      -1.17984    1.02311  -1.153
## native_country_Canada                        -1.45697    0.84348  -1.727
## native_country_China                         -2.64372    0.88690  -2.981
## native_country_Columbia                      -3.76060    1.13990  -3.299
## native_country_Cuba                          -1.77548    0.85952  -2.066
## `native_country_Dominican-Republic`          -3.83866    1.30919  -2.932
## native_country_Ecuador                       -2.69317    1.11137  -2.423
## `native_country_El-Salvador`                 -2.13046    0.93046  -2.290
## native_country_England                       -1.39332    0.87177  -1.598
## native_country_France                        -0.96400    0.94727  -1.018
## native_country_Germany                       -1.73391    0.84363  -2.055
## native_country_Greece                        -1.90732    0.90741  -2.102
```

```
## native_country_Guatemala                       -2.10625    1.10760  -1.902
## native_country_Haiti                           -0.87990    0.96000  -0.917
## `native_country_Holand-Netherlands`            -7.97142  119.24792  -0.067
## native_country_Honduras                        -1.46297    1.41062  -1.037
## native_country_Hong                            -2.27867    1.01266  -2.250
## native_country_Hungary                         -1.13627    1.01167  -1.123
## native_country_India                           -2.40847    0.86331  -2.790
## native_country_Iran                            -1.43832    0.92817  -1.550
## native_country_Ireland                         -0.66015    0.98701  -0.669
## native_country_Italy                           -0.87007    0.86127  -1.010
## native_country_Jamaica                         -1.35499    0.94812  -1.429
## native_country_Japan                           -1.69091    0.89640  -1.886
## native_country_Laos                            -3.12907    1.39178  -2.248
## native_country_Mexico                          -2.23214    0.82786  -2.696
## native_country_Nicaragua                       -2.20456    1.12335  -1.962
## `native_country_Outlying-US(Guam-USVI-etc)`    -2.07957    1.35677  -1.533
## native_country_Peru                            -3.29930    1.15746  -2.850
## native_country_Philippines                     -1.90122    0.84760  -2.243
## native_country_Poland                          -1.39888    0.88177  -1.586
## native_country_Portugal                        -1.37540    0.93662  -1.468
## `native_country_Puerto-Rico`                   -1.95677    0.87148  -2.245
## native_country_Scotland                        -3.80022    1.35591  -2.803
## native_country_South                           -3.57709    0.93179  -3.839
## native_country_Taiwan                          -1.95141    0.93755  -2.081
## native_country_Thailand                        -3.15920    1.18736  -2.661
## `native_country_Trinadad&Tobago`               -2.46061    1.17213  -2.099
## `native_country_United-States`                 -1.58561    0.80228  -1.976
## native_country_Vietnam                         -3.10546    1.02168  -3.040
## native_country_Yugoslavia                            NA         NA      NA
##                                              Pr(>|z|)
## (Intercept)                                  0.590470
## age                                           < 2e-16 ***
## education_num                                 < 2e-16 ***
## capital_gain                                  < 2e-16 ***
## capital_loss                                  < 2e-16 ***
## hours_per_week                                < 2e-16 ***
## `workclass_?`                                      NA
## `workclass_Federal-gov`                      0.153701
## `workclass_Local-gov`                        0.570241
## `workclass_Never-worked`                           NA
## workclass_Private                            0.400106
## `workclass_Self-emp-inc`                     0.320107
## `workclass_Self-emp-not-inc`                 0.888877
## `workclass_State-gov`                        0.644354
## `workclass_Without-pay`                            NA
## marital_status_Divorced                      0.280983
## `marital_status_Married-AF-spouse`           0.000406 ***
## `marital_status_Married-civ-spouse`          1.72e-13 ***
## `marital_status_Married-spouse-absent`       0.810542
## `marital_status_Never-married`               0.000930 ***
## marital_status_Separated                     0.830550
## marital_status_Widowed                             NA
## `occupation_?`                                     NA
## `occupation_Adm-clerical`                    0.260104
```

```
## `occupation_Armed-Forces`                    0.629285
## `occupation_Craft-repair`                    0.134331
## `occupation_Exec-managerial`                  < 2e-16 ***
## `occupation_Farming-fishing`                 1.58e-10 ***
## `occupation_Handlers-cleaners`               7.27e-05 ***
## `occupation_Machine-op-inspct`               0.010375 *
## `occupation_Other-service`                   4.44e-10 ***
## `occupation_Priv-house-serv`                 0.031855 *
## `occupation_Prof-specialty`                  4.91e-12 ***
## `occupation_Protective-serv`                 3.59e-06 ***
## occupation_Sales                             2.36e-05 ***
## `occupation_Tech-support`                    5.50e-08 ***
## `occupation_Transport-moving`                      NA
## relationship_Husband                          < 2e-16 ***
## `relationship_Not-in-family`                 0.036201 *
## `relationship_Other-relative`                4.94e-12 ***
## `relationship_Own-child`                     6.08e-09 ***
## relationship_Unmarried                       0.005200 **
## relationship_Wife                                  NA
## `race_Amer-Indian-Eskimo`                    0.001531 **
## `race_Asian-Pac-Islander`                    0.021311 *
## race_Black                                   0.002957 **
## race_Other                                   0.852313
## race_White                                         NA
## sex_Female                                    < 2e-16 ***
## sex_Male                                           NA
## `native_country_?`                                 NA
## native_country_Cambodia                      0.248832
## native_country_Canada                        0.084109 .
## native_country_China                         0.002875 **
## native_country_Columbia                      0.000970 ***
## native_country_Cuba                          0.038861 *
## `native_country_Dominican-Republic`          0.003367 **
## native_country_Ecuador                       0.015381 *
## `native_country_El-Salvador`                 0.022039 *
## native_country_England                       0.109983
## native_country_France                        0.308841
## native_country_Germany                       0.039851 *
## native_country_Greece                        0.035558 *
## native_country_Guatemala                     0.057218 .
## native_country_Haiti                         0.359373
## `native_country_Holand-Netherlands`          0.946703
## native_country_Honduras                      0.299686
## native_country_Hong                          0.024437 *
## native_country_Hungary                       0.261367
## native_country_India                         0.005274 **
## native_country_Iran                          0.121232
## native_country_Ireland                       0.503598
## native_country_Italy                         0.312389
## native_country_Jamaica                       0.152965
## native_country_Japan                         0.059250 .
## native_country_Laos                          0.024561 *
## native_country_Mexico                        0.007012 **
## native_country_Nicaragua                     0.049707 *
```

```
## `native_country_Outlying-US(Guam-USVI-etc)` 0.125340
## native_country_Peru                        0.004365 **
## native_country_Philippines                 0.024892 *
## native_country_Poland                      0.112638
## native_country_Portugal                    0.141974
## `native_country_Puerto-Rico`               0.024747 *
## native_country_Scotland                    0.005067 **
## native_country_South                       0.000124 ***
## native_country_Taiwan                      0.037399 *
## native_country_Thailand                    0.007798 **
## `native_country_Trinadad&Tobago`           0.035794 *
## `native_country_United-States`             0.048112 *
## native_country_Vietnam                     0.002369 **
## native_country_Yugoslavia                        NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 35452  on 31655  degrees of freedom
## Residual deviance: 20411  on 31575  degrees of freedom
## AIC: 20573
##
## Number of Fisher Scoring iterations: 9
```

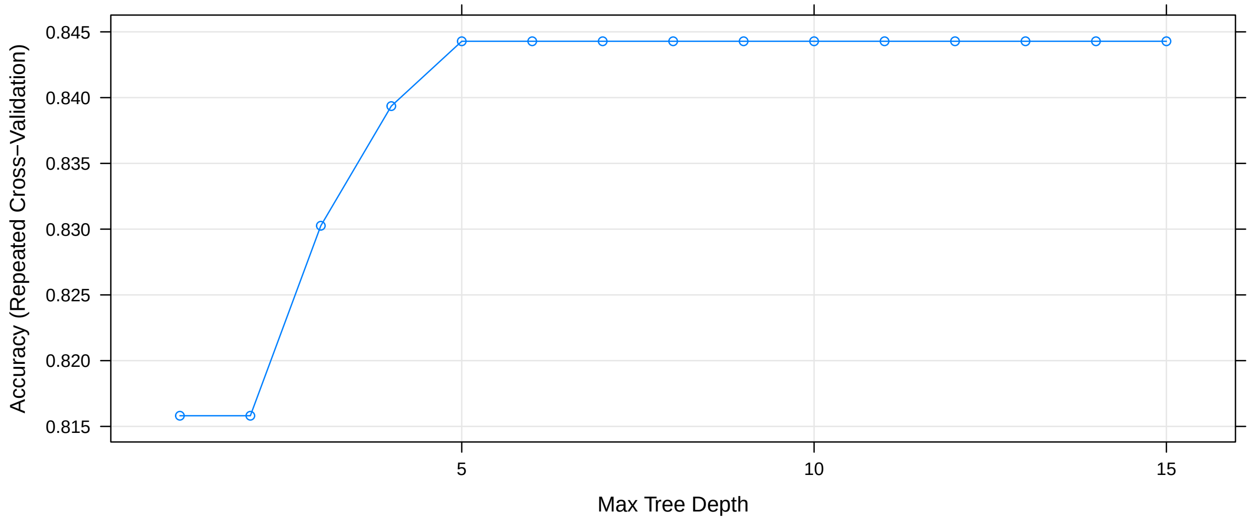
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 9449  755
##          1 1307 2055
##
##                Accuracy : 0.848
##                  95% CI : (0.8418, 0.854)
##     No Information Rate : 0.7929
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5685
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.8785
##             Specificity : 0.7313
##          Pos Pred Value : 0.9260
##          Neg Pred Value : 0.6112
##              Prevalence : 0.7929
##          Detection Rate : 0.6965
##    Detection Prevalence : 0.7522
##       Balanced Accuracy : 0.8049
##
##        'Positive' Class : 0
##
```

Our logistic regression model with dummy variables accuracy comes out as 0.848 or roughly 85%. We can see that this has not changed from the previous two models. There is room for improvement in this model's sensitivity among other variables. We try to improve this with the random forest model.

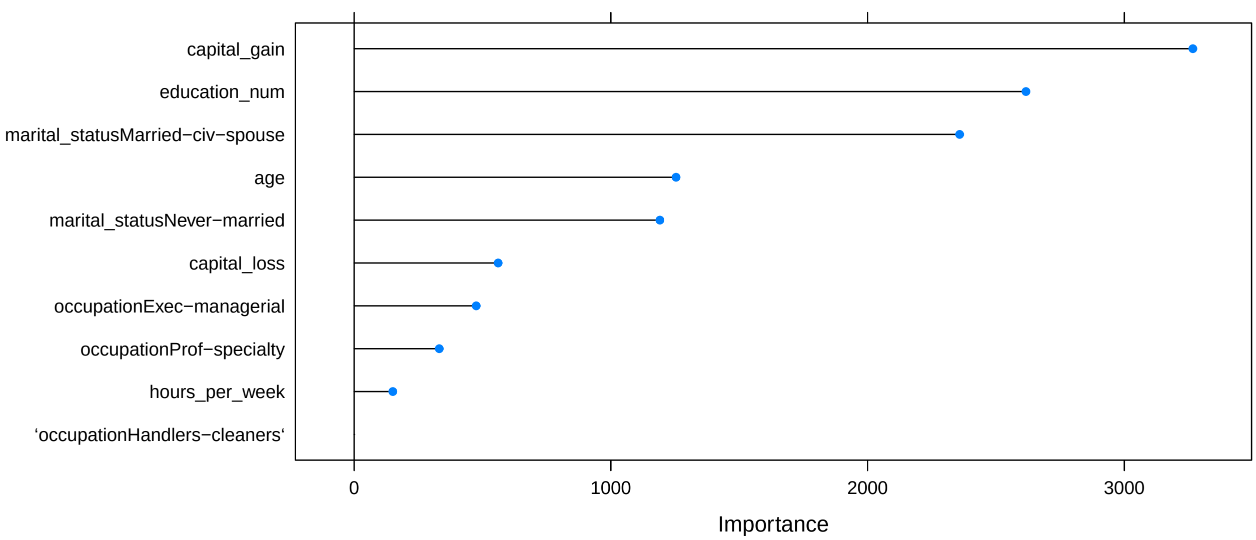
## Decision Trees

In a decision tree model the data is split into distinct options of 'yes' or 'no' based on parameters that make the options possible. These splits are called nodes and the decisions made at them can be mapped. We follow this principle to identify decisions that would result in the most predictive accuracy for our income target variable. The results are shown.

```
## CART
##
## 31656 samples
##    12 predictor
##     2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 28491, 28491, 28490, 28490, 28490, 28491, ...
## Resampling results across tuning parameters:
##
##   maxdepth  Accuracy   Kappa
##    1        0.8158114  0.4303746
##    2        0.8158114  0.4303746
##    3        0.8302584  0.4869948
##    4        0.8393562  0.5216016
##    5        0.8442842  0.5433691
##    6        0.8442842  0.5433691
##    7        0.8442842  0.5433691
##    8        0.8442842  0.5433691
##    9        0.8442842  0.5433691
##   10        0.8442842  0.5433691
##   11        0.8442842  0.5433691
##   12        0.8442842  0.5433691
##   13        0.8442842  0.5433691
##   14        0.8442842  0.5433691
##   15        0.8442842  0.5433691
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was maxdepth = 5.
```




We also review which variables are most important for making decisions in our model.These are shown in the plot as a straight line extending from the axis to the length of its importance to the model. Accuracy was also used to select the optimal model using the largest value where our final tree depth used for this model is 1.




```
## [1] 0.8424001
```

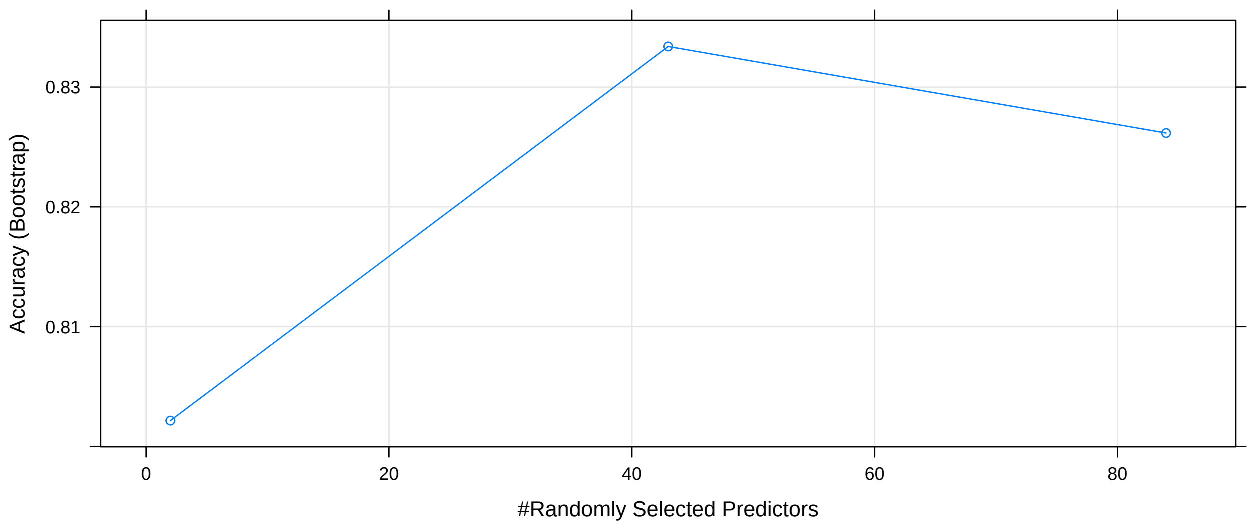
Our decision Tree model accuracy comes out as 0.8424 or roughly 84%. There is room for improvement in this model's sensitivity among other variables. We try to improve this with the random forest model.

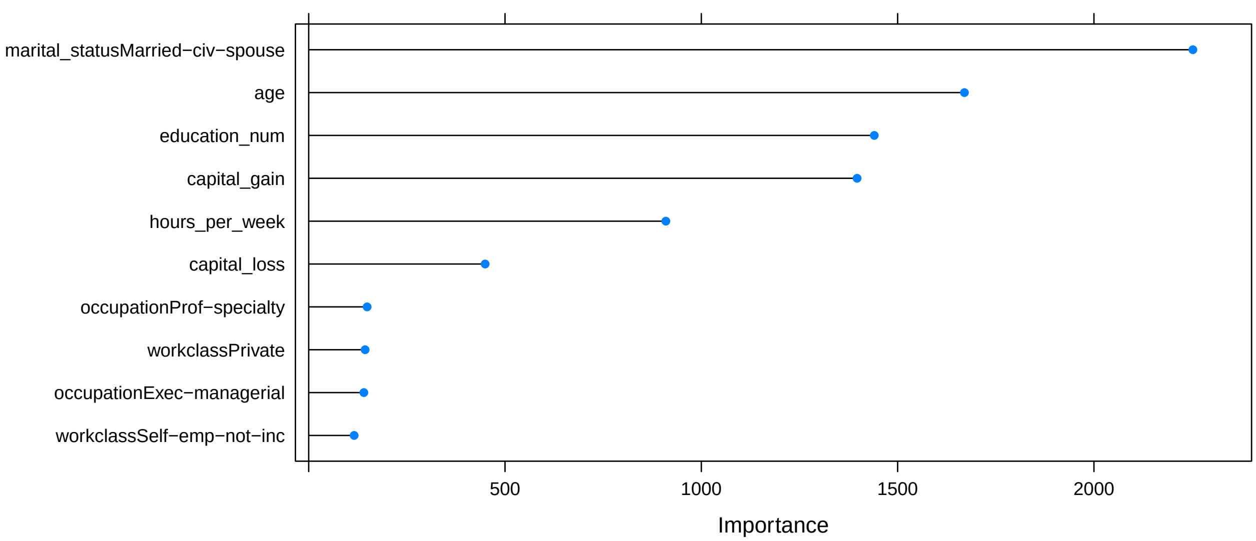
## Random Forests

A random forest model works by building a number of decision trees and selecting the most accurate decisions from the trees. These decisions are randomized and in our case, tries 3 variables at each node or split in the tree. We set our number of trees to 500 and train the model to predict loan status. We review the variables of most importance in the model and in this case, give the model a boost to improve accuracy.

```
## Random Forest
##
## 31656 samples
##    12 predictor
```

```
##     2 classes: '0', '1'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 31656, 31656, 31656, 31656, 31656, 31656, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.8021560  0.3128271
##   43    0.8333873  0.5419169
##   84    0.8261639  0.5256646
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 43.
```



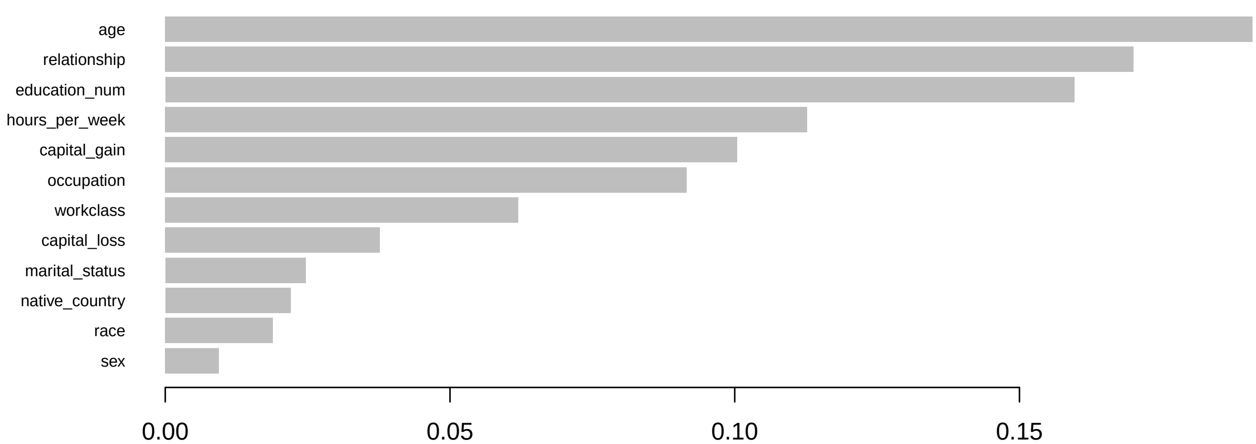

```
## [1] NaN
```

Our random Forest model accuracy comes out as 0.8399 or roughly 84%.. This is an improvement upon our decision model and the sensitivity did increase as we desired. However, there are some more strategies we can try with other models.

#XGboost

```
##    Epoch Iteration max_depth min_child_weight subsample gpUtility acqOptimum
## 1:     0         1         4        16.900949 0.3980034        NA      FALSE
## 2:     0         2         9        22.465545 0.4598973        NA      FALSE
## 3:     0         3         4         2.543344 0.3380428        NA      FALSE
## 4:     0         4         7         8.946161 0.2773560        NA      FALSE
## 5:     1         5         8         1.000000 0.2500000 0.7678954       TRUE
## 6:     2         6        10         1.000000 0.2500000 0.4972389       TRUE
## 7:     3         7        10         1.000000 0.5000000 0.2777107       TRUE
##    inBounds Elapsed     Score nrounds errorMessage
## 1:     TRUE   4.670 0.8944900     100           NA
## 2:     TRUE   8.884 0.9070150     100           NA
## 3:     TRUE   4.277 0.8964657     100           NA
## 4:     TRUE   6.325 0.9071000     100           NA
## 5:     TRUE   7.286 0.9117193      99           NA
## 6:     TRUE   8.524 0.9128047     100           NA
## 7:     TRUE  11.171 0.9130480     100           NA
```

```
## $max_depth
## [1] 10
##
## $min_child_weight
## [1] 1
##
## $subsample
## [1] 0.5
```




```
## Confusion Matrix and Statistics
##
##            y_test
## XGB.predict    0    1
##           0 9317 1209
##           1  887 2153
##
##                Accuracy : 0.8455
##                  95% CI : (0.8393, 0.8515)
##     No Information Rate : 0.7522
##     P-Value [Acc > NIR] : < 2.2e-16
```

```
##
##                   Kappa : 0.5718
##
##  Mcnemar's Test P-Value : 2.358e-12
##
##             Sensitivity : 0.9131
##             Specificity : 0.6404
##          Pos Pred Value : 0.8851
##          Neg Pred Value : 0.7082
##              Prevalence : 0.7522
##          Detection Rate : 0.6868
##    Detection Prevalence : 0.7759
##       Balanced Accuracy : 0.7767
##
##        'Positive' Class : 0
##
```

# Model Performance

When evaluating how the models performed we focused on accuracy as our main metric. However, we also considered how the results might apply in real-world settings. This does slightly change the results of our model's performances depending on the circumstances in which the prediction is needed. For example, someone with the goal of identifying what factors they need to maximize to boost their income will have a fundamentally different set of variables, and thus results, than someone else with the goal of minimizing income loss for an individual. Nevertheless we compiled the results as follows:

| statistic | xgboost | decisiontree | randomforest | logit |
|---|---|---|---|---|
| Accuracy | 0.845 | 0.842 | 0.840 | 0.848 |
| Kappa | 0.572 | 0.542 | 0.559 | 0.569 |
| AccuracyLower | 0.839 | 0.836 | 0.834 | 0.842 |
| AccuracyUpper | 0.852 | 0.848 | 0.846 | 0.854 |
| AccuracyNull | 0.752 | 0.752 | 0.752 | 0.793 |
| AccuracyPValue | 0.000 | 0.000 | 0.000 | 0.000 |
| McnemarPValue | 0.000 | 0.000 | 0.000 | 0.000 |

# Conclusion

Our conclusion could be given regardless of model performance and accuracy given the diversity of the dataset and its substantial drawbacks. Perhaps most importantly, we should note that this dataset was not representative of the global population and should not be applied too broadly. This dataset was heavily white, highly educated males who were married at least once in their lives. Many of these respondents also had no kids which evidence suggest can significantly shape an individual's income over their lifetime. Typically, having kids increases income for males while it decreases for females. This makes our results less realistic and hard to interpret, especially for non-white females and other minority classes not represented in this dataset.

Additionally, responses from individuals located in the U.S. dominated the list, containing nearly 90% of the dataset's individuals. This nullifies the results for other countries due to large clusters of outliers in their variables that could not be dealt with without comprising the integrity of the data. To reduce the errors inherent to the dataset, an extensive use of oversampling of the minority classes in a strategic manner would be necessary but unfortunately, there is no way to tell if the results would be reliable. For these reasons, we focus on the relationships between the variables which have greater reliability and certainty in this analysis.

Recall that our target variable, income, was split into two factor levels; those whose income is greater than $50,000 and those who have an income less than or equal to $50,000. As is, our XGBoosted model performed best with an accuracy between 82-87%. Our closest alternative model was the random forest. Excluding capital gain and losses, we found that age, education, and the hours worked per week, capture nearly perfectly the variance in the dataset. If we were to reduce the dimensions of the dataset these would be the best variables to use. This suggests that, aside from capital gains, the best ways to increase income to $50,000 or greater in the United States is to get a higher education, work 40 or more hours per week and be older than your colleagues. These results are applicable across the United States.

# References


https://archive.ics.uci.edu/ml/datasets/Census+Income https://stats.idre.ucla.edu/r/dae/logit-regression/ https://www.datacamp.com/community/tutorials/logistic-regression-R https://www.marsja.se/create-dummy-variables-in-r/


# Code Appendix

```
knitr::opts_chunk$set(echo=FALSE, error=FALSE, warning=FALSE, message=FALSE, fig.align="center", fig.wi
# Libraries
library(dplyr)
library(summarytools)
library(reshape2)
library(ggplot2)
library(DataExplorer)
library(caret)
library(tidyverse)
library(DataExplorer)
library(mice)
library(MASS)
library(e1071)
library(tree)
library(randomForest)
library(corrplot)
library(kableExtra)
library(htmltools)
library(fastDummies)
library(mlbench)
library(xgboost)
library(ParBayesianOptimization)
library(factoextra)
income_data <- read.csv("https://raw.githubusercontent.com/amit-kapoor/Data622Group2/main/FinalProject/c
                        check.names = FALSE) %>%
  na_if("")

# categorical columns as factors
income_data <- income_data %>%
  mutate(workclass=as.factor(workclass),
         education=as.factor(education),
         marital_status=as.factor(marital_status),
         occupation=as.factor(occupation),
         relationship=as.factor(relationship),
         race=as.factor(race),
         sex=as.factor(sex),
```

```r
         native_country=as.factor(native_country),
         income=as.factor(income))

dfSummary(income_data, style = 'grid', graph.col = FALSE)
summary(income_data)
income_data %>%
  count(income) %>%
  ggplot(data=., aes(x=factor(income), y=n, fill = income)) +
  geom_col() +
  xlab("Income") +
  ylab("Frequency") +
  ggtitle("Frequency of Income") +
  theme_classic() +
  theme(legend.position = "none")


# select categorical columns
cat_cols = c()
j <- 1
for (i in 1:ncol(income_data)) {
  if (class((income_data[,i])) == 'factor') {
      cat_cols[j]=names(income_data[i])
      j <- j+1
  }
}

income_fact <-  income_data[cat_cols]
# long format
income_factm <- melt(income_fact, measure.vars = cat_cols, variable.name = 'metric', value.name = 'valu

# plot categorical columns
ggplot(income_factm, aes(x = value)) +
  geom_bar(aes(fill = metric)) +
  facet_wrap( ~ metric, nrow = 5L, scales = 'free') + coord_flip() +
  theme_classic() +
  theme(legend.position = "none")
plot_histogram(income_data, geom_histogram_args = list("fill" = "tomato4"))
ggplot(income_data, aes(x=native_country, fill=income)) +
  geom_bar() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
ggplot(income_data, aes(x=workclass, fill=income)) +
  geom_bar() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
ggplot(income_data, aes(x=education, fill=income)) +
  geom_bar(stat = "count") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
ggplot(income_data, aes(x=sex, fill=income)) +
  geom_bar() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
ggplot(income_data, aes(x=race, fill=income)) +
  geom_bar() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
cors <- income_data %>%
```

```r
  select_if(is.numeric) %>%
  na.omit() %>%
  cor()
corrplot::corrplot(cors, method="number")

index <- income_data == "?"
is.na(income_data) <- index
# plot missing values
plot_missing(income_data)
# finding NAs now in income_data
sum(!complete.cases(income_data))
income_data_clean <- income_data[complete.cases(income_data),]
dim(income_data_clean)
library(tidyr)
df <- income_data_clean
df %>%
  dplyr::select_if(is.integer) %>%
  gather(key, value) %>%
  ggplot(aes(key, value)) +
  geom_boxplot(aes(fill = key)) +
  facet_wrap(~key, scales = "free") # Lots of outliers
# removing columns fnlwgt and education
income_data_clean <- income_data_clean %>%
  dplyr::select(-c(fnlwgt, education))
set.seed(622)

# Center and scaling for numeric features
income_data_tf <- income_data_clean %>%
  dplyr::select(c("age", "education_num" , "capital_gain", "capital_loss","hours_per_week")) %>%
  preProcess(method = c("BoxCox", "center","scale")) %>%
  predict(income_data_clean)
income_data_tf$income <- plyr::mapvalues(income_data_tf$income, from = c('>50K','<=50K'), to = c(1,0))
head(income_data_tf)
nums <- income_data_clean %>%
  dplyr::select(is.numeric)
pca1 <- prcomp(nums)
summary(pca1)
plot(pca1, type = 'l', col = 'light blue')
fviz_eig(pca1)
fviz_contrib(pca1, choice = "var", axes = c(1,2), top = 15)
fviz_pca_var(pca1,
             col.var ="contrib", # Color by contributions to the PC
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE     # Avoid text overlapping
             ,axes=c(1,2)
             )
# PCA with removal of capital_gains and losses
nums2 <- income_data_clean %>%
  dplyr::select(is.numeric, -capital_gain, - capital_loss)
pca2 <- prcomp(nums2)
summary(pca2)
fviz_eig(pca2)
fviz_contrib(pca2, choice = "var", axes = c(1,2), top = 15)
```

```r
fviz_pca_var(pca2,
             col.var ="contrib", # Color by contributions to the PC
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE     # Avoid text overlapping
             ,axes=c(1,2)
             )
facs <- income_data_clean %>%
  dplyr::select(is.factor)
facs_nums <- sapply(facs, as.numeric)
dfnumeric <- cbind(facs_nums, nums)
pca3 <- prcomp(dfnumeric)
prcomp <- prcomp(dfnumeric, scale. = TRUE, center=TRUE)
summary(pca3)
fviz_eig(pca3)
fviz_contrib(pca3, choice = "var", axes = c(1,2), top = 15)
fviz_pca_var(pca3,
             col.var ="contrib", # Color by contributions to the PC
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE     # Avoid text overlapping
             ,axes=c(1,2)
             )
dfnumeric2 <- dfnumeric %>%
  dplyr::select(-capital_gain, -capital_loss)
pca4 <- prcomp(dfnumeric2)
summary(pca4)
fviz_eig(pca4)
fviz_contrib(pca4, choice = "var", axes = c(1,2), top = 15)
fviz_pca_var(pca4,
             col.var ="contrib", # Color by contributions to the PC
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE     # Avoid text overlapping
             ,axes=c(1,2)
             )
set.seed(622)
partition <- createDataPartition(income_data_tf$income, p=0.70, list = FALSE)
training <- income_data_tf[partition,]
testing <- income_data_tf[-partition,]
logit.income <-  glm(income ~., data = training, family = "binomial")
summary(logit.income)
logit.pred <- predict(logit.income, testing, type="response")
testing$pred_glm <- ifelse(logit.pred > 0.5, "1", "0")
testing$pred_glm <- as.factor(testing$pred_glm)
testing$income <- as.factor(testing$income)
conf.mat.logit <- confusionMatrix(testing$income, testing$pred_glm)
conf.mat.logit
logit.income.nc <-  glm(income ~age + workclass + education_num + marital_status + occupation +
    relationship + race + sex + capital_gain + capital_loss +
    hours_per_week, data = training, family = "binomial")
summary(logit.income.nc)
logit.pred.nc <- predict(logit.income.nc, testing, type="response")
testing$pred_glm2 <- ifelse(logit.pred > 0.5, "1", "0")
testing$pred_glm2 <- as.factor(testing$pred_glm2)
testing$income <- as.factor(testing$income)
```

```
conf.mat.logit.nc <- confusionMatrix(testing$income, testing$pred_glm2)
conf.mat.logit.nc

library(fastDummies)
set.seed(622)
training.dum <- dummy_cols(training,
                           select_columns = c("workclass",
                                              "marital_status",
                                              "occupation",
                                              "relationship",
                                              "race",
                                              "sex",
                                              "native_country"),
                           remove_selected_columns = TRUE)

set.seed(622)
testing.dum <- dummy_cols(testing,
                           select_columns = c("workclass",
                                              "marital_status",
                                              "occupation",
                                              "relationship",
                                              "race",
                                              "sex",
                                              "native_country"),
                           remove_selected_columns = TRUE)

logit.income.dum <- glm(income ~., data = training.dum, family = "binomial")
summary(logit.income.dum)
logit.pred.dum <- predict(logit.income.dum, testing.dum, type="response")
testing.dum$pred_glm <- ifelse(logit.pred.dum > 0.5, "1", "0")
testing.dum$pred_glm <- as.factor(testing.dum$pred_glm)
testing.dum$income <- as.factor(testing.dum$income)
conf.mat.logit.dum <- confusionMatrix(testing.dum$income, testing.dum$pred_glm)
conf.mat.logit.dum
# Check Number of Levels for each Factor
training %>% map(levels) %>% map(length)
testing %>% map(levels) %>% map(length)
# Decision Trees model
set.seed(622)
control <- trainControl(method="repeatedcv", number=10, repeats=3, search='grid')
metric <- "Accuracy"
tunegrid <- expand.grid(.maxdepth=c(1:15))
tree.income <- train(income~., data = training, method="rpart2", tuneGrid=tunegrid, trControl=control)
print(tree.income)
plot(tree.income)
treeImp <- varImp(tree.income, scale = FALSE)
plot(treeImp, top = 10)
# prediction from decision tree model
tree.predict <- predict(tree.income, testing,type='raw')
mean(tree.predict == testing$income) # accuracy
conf.mat.decisiontree <- confusionMatrix(tree.predict, testing$income)
var1 <- round(conf.mat.decisiontree$overall[1], 4)
var2 <- (round(conf.mat.decisiontree$overall[1], 2)*100)
```

```r
set.seed(622)
# Random Forest model
rf.income <- train(income~., data = training, method="rf", ntree = 5)
print(rf.income)
plot(rf.income)
rfImp <- varImp(rf.income, scale = FALSE)
plot(rfImp, top = 10)
# prediction from random forest model
rf.predict <- predict(rf.income, testing,type='raw')
mean(rf.predict == testing$Loan_Status) # accuracy
conf.mat.randomforest <- confusionMatrix(rf.predict, testing$income)
y_train <- as.matrix(training$income)
y_test <- as.numeric(as.matrix(testing$income))
X_train <- sapply(subset(training, select = -income), as.numeric)
X_test <- sapply(subset(testing, select = -c(income, pred_glm, pred_glm2)), as.numeric)
Folds <- list(
    Fold1 = as.integer(seq(1,nrow(X_train),by = 3))
  , Fold2 = as.integer(seq(2,nrow(X_train),by = 3))
  , Fold3 = as.integer(seq(3,nrow(X_train),by = 3))
)

scoringFunction <- function(max_depth, min_child_weight, subsample) {
  dtrain <- xgb.DMatrix(X_train, label=y_train)
  Pars <- list(
      booster = "gbtree"
    , eta = 0.01
    , max_depth = max_depth
    , min_child_weight = min_child_weight
    , subsample = subsample
    , objective = "binary:logistic"
    , eval_metric = "auc"
  )
  xgbcv <- xgb.cv(
      params = Pars
    , data = dtrain
    , nround = 100
    , folds = Folds
    , prediction = TRUE
    , showsd = TRUE
    , early_stopping_rounds = 5
    , maximize = TRUE
            , verbose = 0)
  return(
    list(
        Score = max(xgbcv$evaluation_log$test_auc_mean)
      , nrounds = xgbcv$best_iteration
    )
  )
}

set.seed(50)
bounds <- list(
    max_depth = c(2L, 10L)
```

```r
  , min_child_weight = c(1, 25)
  , subsample = c(0.25, .5)
)

optObj <- bayesOpt(
    FUN = scoringFunction
  , bounds = bounds
  , initPoints = 4
  , iters.n = 3
)
optObj$scoreSummary
print(getBestPars(optObj))

dt <- xgb.DMatrix(X_train, label=y_train)
XGB <- xgboost(data = dt
    , nround = 100
    , min_child_weight=1
    , subsample=.5
    , max_depth = 10
    , early_stopping_rounds = 5
            , verbose = 0)

XGB.predict <- as.numeric(predict(XGB,X_test) > 0.5)

importance_matrix <- xgb.importance(model = XGB)

xgb.plot.importance(importance_matrix = importance_matrix)
conf.mat.xgboost <- confusionMatrix(table(XGB.predict, y_test))
print(conf.mat.xgboost)
results <- data.frame(matrix(names(conf.mat.xgboost$overall)))
results$xgboost <- round(conf.mat.xgboost$overall, 3)
results$decisiontree <- round(conf.mat.decisiontree$overall, 3)
results$randomforest <- round(conf.mat.randomforest$overall, 3)
results$logit <- round(conf.mat.logit$overall, 3)

results %>%
  rename(statistic = matrix.names.conf.mat.xgboost.overall..) %>%
  kable()
```