Predicting Income using U.S. Census Data and Classification Algorithms

Abstract: This project aims at analyzing data of the U.S. Census Bureau and extrapolating predictions on income via machine learning algorithms. The aim is to predict whether an individual's income is greater than $50.000 per year based on attributes from the census data.

Introdution:

The U.S. Census dataset utilized for this project is composed of 48.846 entities, each containing information in the form of 15 columns. The dataset is extracted from the 1994 US Census database.

The columns are as follows:

- **age** : the age of an individual
  - Integer greater than 0
- **workclass** : a general term to represent the employment status of an individual
  - Private, Selfempnotinc, Selfempinc, Federalgov, Localgov, Stategov, Withoutpay, Neverworked.
- **fnlwgt** : final weight. In other words, this is the number of people the census believes the entry represents..
  - Integer greater than 0
- **education** : the highest level of education achieved by an individual.
  - Bachelors, Somecollege, 11th, Hsgrad, Profschool, Assocacdm, Assocvoc, 9th, 7th8th, 12th, Masters, 1st4th, 10th, Doctorate, 5th6th, Preschool.
- **educationnum**: the highest level of education achieved in numerical form.
  - Integer greater than 0
- **maritalstatus**: marital status of an individual. Marriedcivspouse corresponds to a civilian spouse while MarriedAFspouse is a spouse in the Armed Forces.
  - Marriedcivspouse, Divorced, Nevermarried, Separated, Widowed, Marriedspouseabsent, MarriedAFspouse.
- **occupation** : the general type of occupation of an individual
  - Techsupport, Craftrepair, Otherservice, Sales, Execmanagerial, Profspecialty, Handlerscleaners, Machineopinspct, Admclerical, Farmingfishing, Transportmoving, Privhouseserv, Protectiveserv, ArmedForces.
- **relationship** : represents what this individual is relative to others. For example an individual could be a Husband. Each entry only has one relationship attribute and is somewhat redundant with marital status. We might not make use of this attribute at all
  - Wife, Ownchild, Husband, Notinfamily, Otherrelative, Unmarried.
- **race** : Descriptions of an individual's race
  - White, AsianPacIslander, AmerIndianEskimo, Other, Black.
- **sex** : the biological sex of the individual
  - Male, Female

● **capitalgain**: capital gains for an individual
○ Integer greater than or equal to 0
● **capitalloss**: capital loss for an individual
○ Integer greater than or equal to 0
● **hoursperweek**: the hours an individual has reported to work per week
○ continuous.
● **nativecountry**: country of origin for an individual
○ UnitedStates, Cambodia, England, PuertoRico, Canada, Germany, OutlyingUS(GuamUSVIetc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, DominicanRepublic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, ElSalvador, Trinadad&Tobago, Peru, Hong, HolandNetherlands.
● **the label** : whether or not an individual makes more than $50,000 annually.
○ <=50k, >50k

In making our data suitable for input for ML algorithms, the “target” will be the last column, which is composed of binary values (True if >50K, False if <=50K), hence making our task a binary classification problem.

Methodology - Results:
First Look at the Data:

First we examine the data with simple distribution charts to gain insight into what could be the most useful features to use:

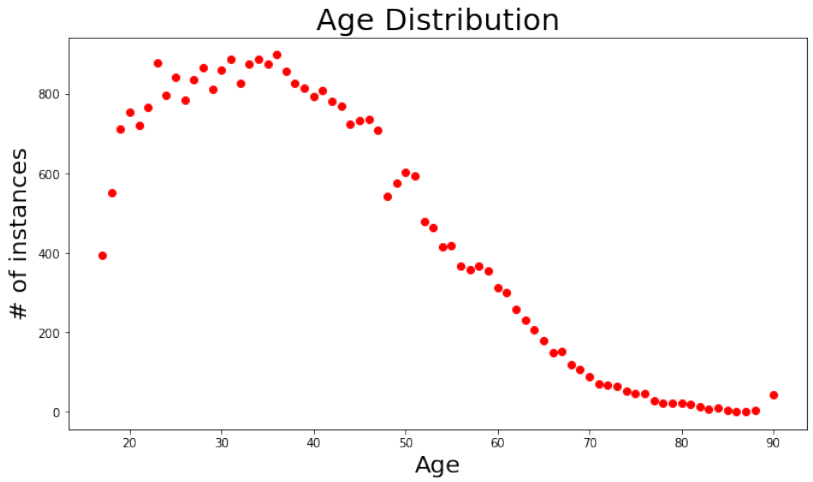

Figure 1: Age Distribution

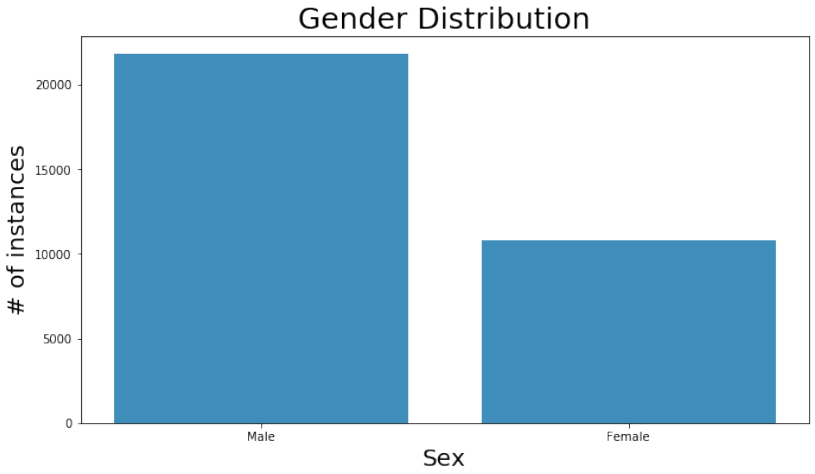

Figure 2: Gender Distribution

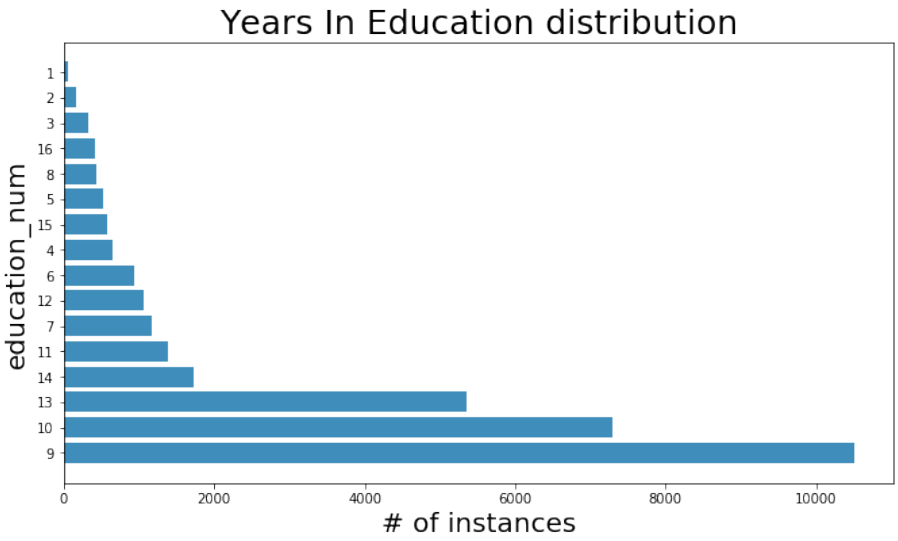

Figure 3: Number of years in education distribution

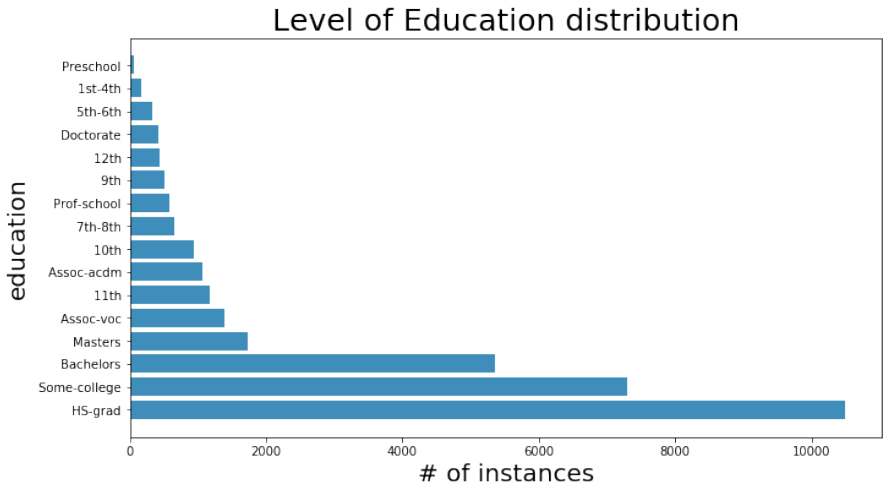

Figure 4: Level of Education distribution

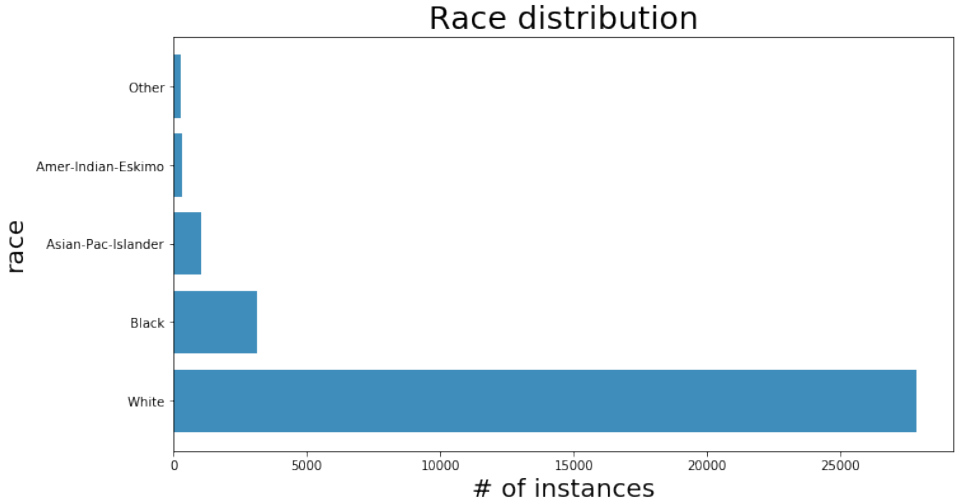

Figure 5: Distribution of race

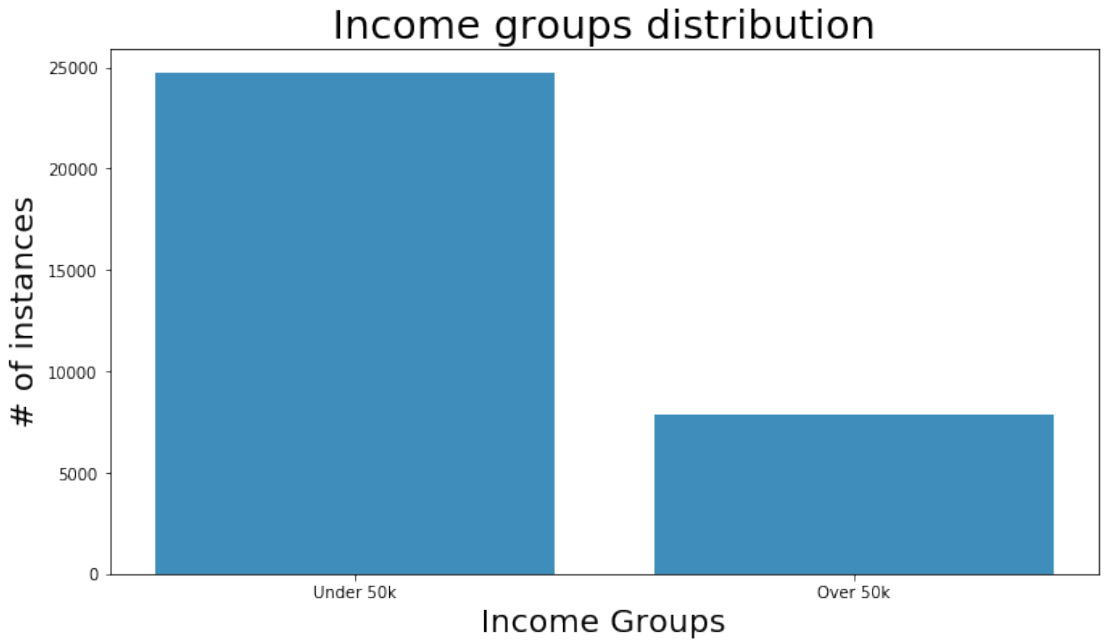

Figure 6: Distribution of income groups

Figure 6 gives us a very important insight into our data; roughly 30% of the citizens gain over 50k a year. Hence, the dataset is unbalanced, meaning the labels are biased towards the “Under 50k” value, which translates to “False”. Although not pictured here, this is true in both datasets (training and test).

Choosing Features:

In this part we preform Logistic Regression on 3 data columns: age, years in education and hours per week, hoping to find some sort of decision boundary which would hint towards a good choice of features for training.

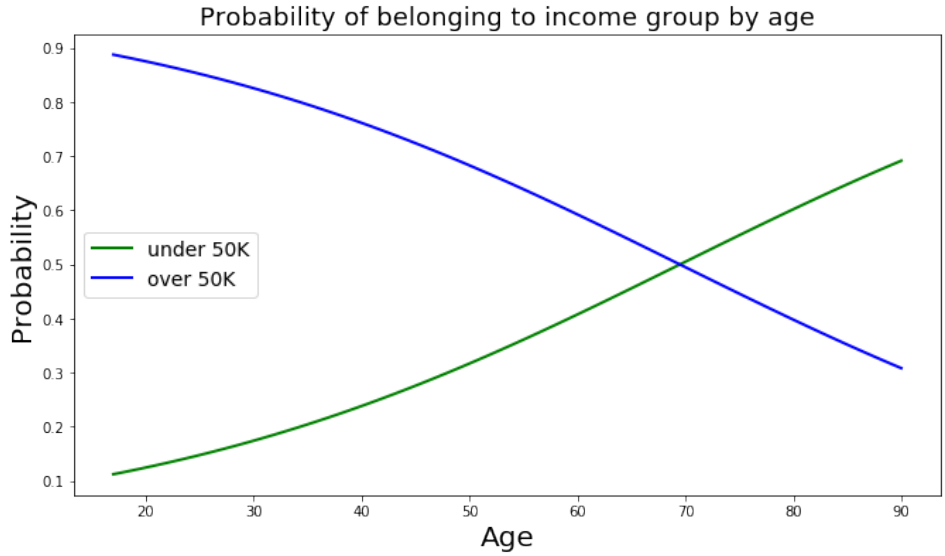

Figure 7: Probability of belonging to income group by age

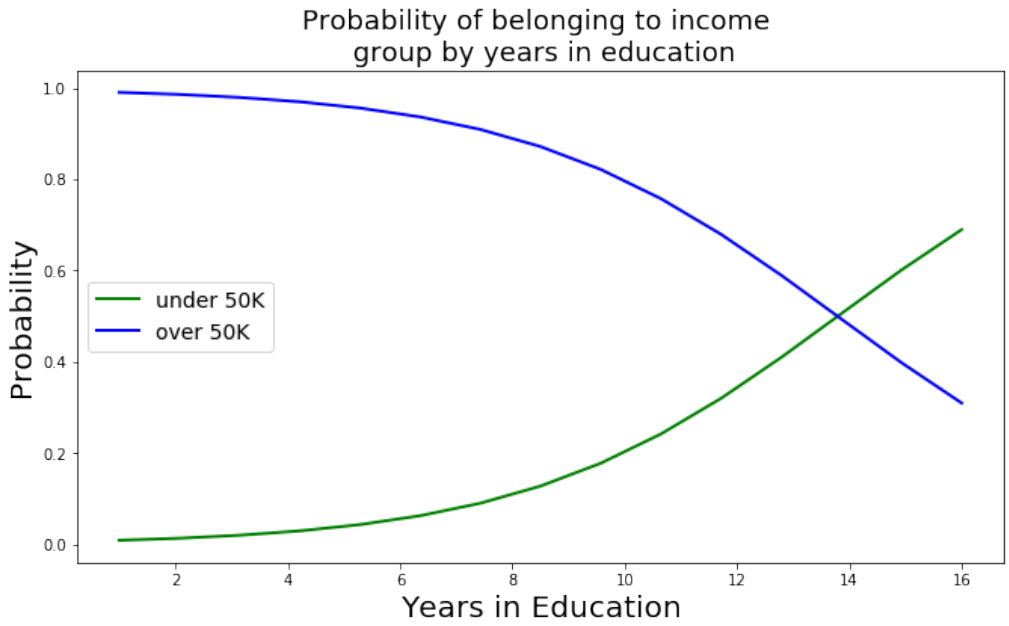

Figure 8: Probability of belonging to income group by years spent in education

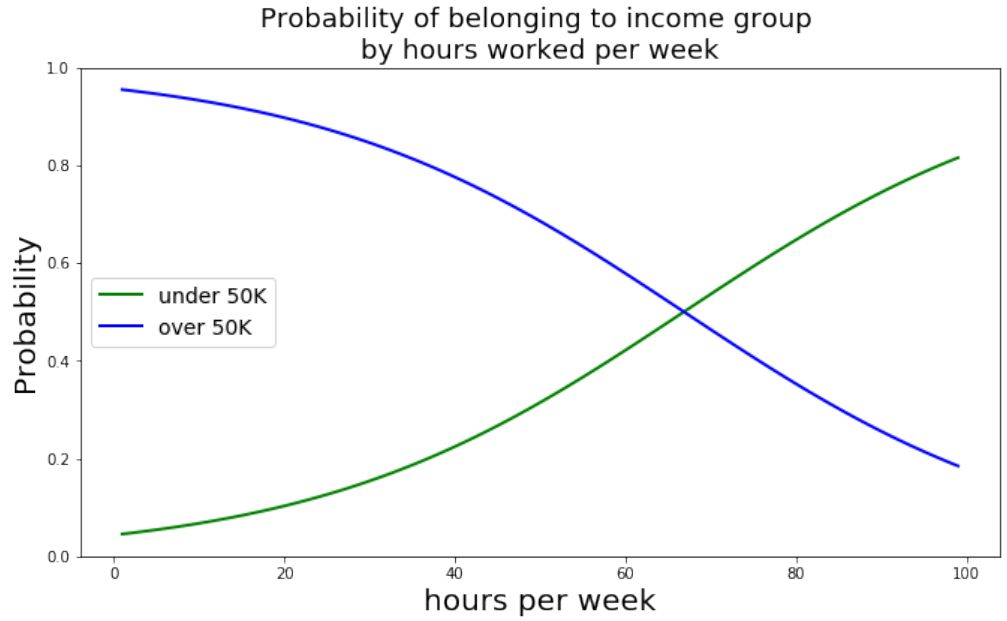

Figure 9: Probability of belonging to income group by hours worked per week

Figures 7, 8 and 9 show us that for each of these features there is a clear decision boundary. These are the features we are now going to use for training.
The features were also selected based on their type, the three we are going to use now are the only useful numerical features, while the rest are composed of string type values (e.g. nativecountry, occupation etc). This could be combated by replacing their values with numerical values based on their corresponding probability of belonging to the “Over 50k” class.
Also, the non-numerical features could be utilized in the Decision Trees method but not in most other methods.

Classification:

In this part we apply different classification methods to try and find the one with the most efficiency.
Accuracy will be measured using cross validation on the training dataset and accuracy of prediction on the test dataset.

Logistic Regression Classifier:

	Cross Validation	Test Accuracy
Accuracy	78.8%	87.4%

Stochastic Gradient Descent Classifier:

	Cross Validation	Test Accuracy
Accuracy	77.6%	55.8%

Decision Trees:

To determine the optimal depth of the Decision Tree, we estimated the cross validation accuracy for different depth values:

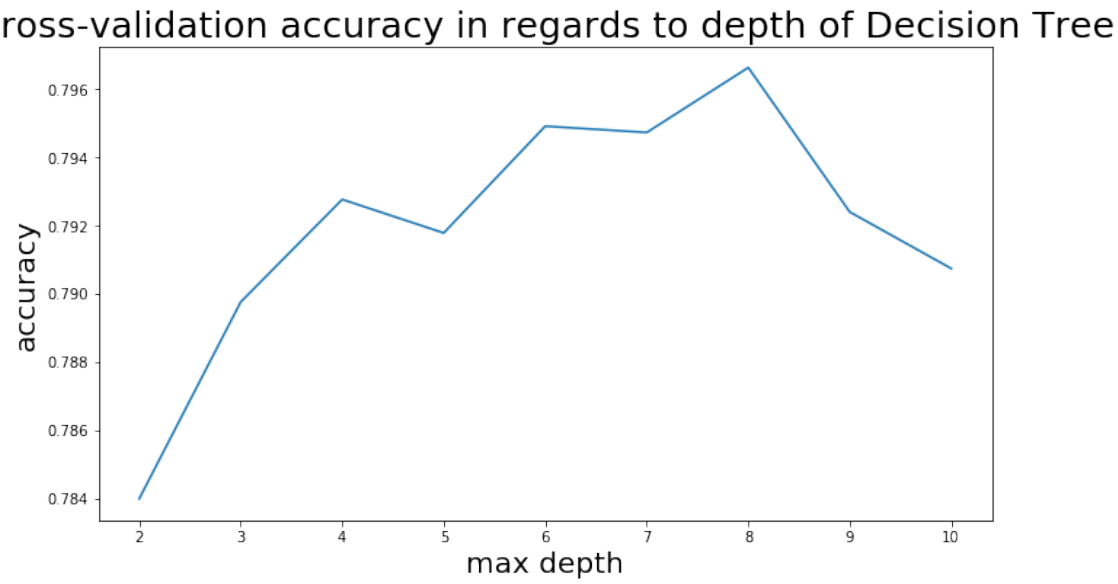

Figure 10: Accuracy in regards to depth

Figure 10 tells us that a Decision Tree with a maximum depth of 8 is the most accurate. Training our this model on our dataset we get:

	Cross Validation	Test Accuracy
Accuracy	79.7%	86.6%

Ensemble Learning Classifiers:

Random Forest:

	Cross Validation	Test Accuracy
Accuracy	77.4%	82.8%

Bagging Classifier:

	Cross Validation	Test Accuracy
Accuracy	79.7%	86.3%

AdaBoost:

	Cross Validation	Test Accuracy
Accuracy	79.9%	85.9%

Gradient Boosting:

	Cross Validation	Test Accuracy
Accuracy	79.9%	85.9%

Voting Classifier:

	Cross Validation	Test Accuracy
Accuracy	79.2%	85.5%

Conclusions:

The best method is a simple Logistic Regression Classifier. This is likely due to the binary nature of the classification task (2 classes: True / False), and Logistic Regression's sigmoid function's efficiency in binary classification.

We conclude that ensemble learning algorithms generally achieve similar accuracy.

Generally, even the best accuracy score we get is relatively unsatisfactory. This brings us to necessary future changes that we estimate will bring us closer to 100% accuracy.

Notes on Improvement:

- More features: An obvious improvement would be the utilization of more of the features via the process described above: replacing string values with numerical values based on their corresponding probability of belonging to the “Over 50k” class.

-Baselinining the data: As described above, rougly 1/3 of the data belongs to one of two classes. Moreover, that ratio is slightly different between the test and train dataset - hence the difference in cross validation accuracy and test accuracy -. By bringing the ratio closer to 1, we could get better accuracy on of the both datasets.

Related Literature:
The methods used in this project are examined in the book Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition by Aurélien Géron.
[https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/]

The U.S. Census Database was extracted from the census bureau database found at [http://www.census.gov/ftp/pub/DES/www/welcome.html] and is available at [https://www.kaggle.com/uciml/adult-census-income]