

Introduction

This is a data mining exercise, performed by Mohamed and Jacob, to use statistical computing to solve a classification problem. The dataset in which we will be performing analysis/prediction can be found on the UCI Machine Learning Repository web site [Census Income Data Set](#).

This particular dataset, compiled from census data, contains 14 attributes (listed below) defining a person's demographic, behavioral and socioeconomic characteristics in an attempt to predict income levels- above or below a threshold of \$50k per year. The data set is robust, comparatively speaking, containing 48,842 instances. In addition, the dataset also includes a test dataset in which we are planning to use in our models to crossvalidate.

Attribute Definitions:

Attributes are either categorical or integers, however, some attributes contain missing values within both datasets (training and test).

Class: >50K, <=50K.

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.

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, Trinidad&Tobago, Peru, Hong, Holand-Netherlands.

Of these attributes, `fnlwgt` was indecipherable to a person without Census domain knowledge. It is an abbreviation for finalweight, and acts as a measurement of similarity between two unique individuals, in regards to economic and behavioral characteristics. It should be noted that `fnlwgt` is a relevant metrics, if and only if it is used to compare individuals within the same state- “People with similar demographic characteristics should have similar weights. There is one important caveat to remember about this statement. That is, since the CPS sample is actually a collection of 51 state samples, each with its own probability of selection, the statement only applies within state”  [Link](#).

Initial Strategies:

There are a wide range of techniques that we can employ to build our models using both R and Weka (We will use both to contrast performance) in the following steps:

- We will use Association Rules Mining to see if any particular attributes are highly correlated with income (Setting the RHS to Income), and evaluate performance using support, confidence and lift. This kind of analysis would be useful if someone were using Census data to target specific individuals, but was unsure of their income level- For instance, we could see that a male with a doctoral degree is likely to have an income exceeding 50k, or Income YES.
- We will also perform cluster analysis for audience segmentation and outlier detection, using both the Simple K Means algorithm, as well as the Hierarchical Clustering Algorithm (We will build models using both Single Linkage, or min distance between two clusters, and complete linkage, which is max distance between clusters. We will also want to cover the distance measures and similarity measure calculations use to derive our clusters.
- The plan is to utilize **Decision Tree theory** for help in predicting classification per the training data set (induction/deduction). Our root nodes will be our attributes, as defined above, and our leaf nodes will be our decisions. In class we have focused heavily on the **J48 algorithm**, but we may also experiment with the **C4.5** tree. Through this analysis, we will derive entropy and **information gain/gain ratio** as well as **Gini and Error** for our attributes, in order to measure importance (Similar to P Values in a regression problem). When building decision tree models, we will employ vast parameter refinements/tuning (Confidence Factor, minNumObj, Reduced Error Pruning, Binary Split, Subtree Raising) in an attempt to create the model with the best performance. Our model evaluation will focus on accuracy, precision, recall and f measure as we perform a variation of holdout tests and N Fold Cross Validation models and compare results. **Our goal** will be to have a model that performs better than both random guess (50/50) and

majority vote (N/48,842). We will document our results via excel and note any changes made to models. We will also experiment with adjustments to our random seeds and take the means/standard deviations of our accuracy per those adjustments.

- Lastly (Up to this point) we will implement measures of Naive Bayes, or Bayesian Theory, in an attempt to solve our classification problem. Naive Bayes focuses on instance probability, and co-occurrence, or conditional, probability. We will look at the variable of Class as an event that is dependent on our other variables. Initial munging is incomplete at this point, so we will need to better understand our continuous variables and decide if smoothing is needed or not. Results will be compiled and compared in a similar manner to our decision tree findings.

Bayes' Theorem:

Bayes' theorem is a mathematical formula for determining conditional probability. To calculate the probability of B given A, the algorithm counts the number of cases where A and B occur together and divides it by the number of cases where A occurs alone.

$$\text{Prob}(B \text{ given } A) = \text{Prob}(A \text{ and } B) / \text{Prob}(A)$$

Our goal is to explore Naive Bayes to calculate the probability of B given A, the algorithm counts the number of cases where A and B occur together and divides it by the number of cases where A occurs alone. We are aiming to detect if Naive-Bayesian classifiers are robust to irrelevant attributes, and classification takes into account evidence from many attributes to make the final prediction.