CMSC 12300 Final Project

Andy Liu and Nelson Auner

June 3, 2013

1 Introduction

Our project investigates the use of census data to predict whether a household self-reports that its income is over or under \$50,000. Our goals, put succintly, were to

1. select a subset of prediction methods to examine, based on interpretability of results and computational feasability
2. test relevant models and select the best one, and,
3. develop an accurate estimation of our model's prediction of a similar, yet different data set.

To this end, we used R and Python, deployed on personal computers as well as AWS, to implement logistical regression of various combinations of regressors. We took the best models, as defined by their performance within the training set, and subjected them to cross-validation. To deal with missing data, we utilized k nearest neighbors to "fill in" missing variables with the nearest closest observation.

Our results suggest the following: Education as a stand-alone predictor is a very good model, with around 5.4% false positive rate and .5% false negative rate. By very good, however, we mean compared to other non-trivial models, considering that the relative lack of diversity in our responses, given an absolute error weighting, favors a classification scheme with all-negative predictions.

2 Description of the Data set

Our analysis is on the Census-Income (KDD) Data Set, a classic machine learning dataset of census data from the 1994 and 1995 Current Population Surveys conducted by the U.S. Census Bureau.

The actual data is made freely available from the UC-Irvine Machine Learning Repository, and consists of 46 variables (see the appendix for a full list), with features such as race, capital gains, country of birth of parents, and weeks worked in the year. Because the census data is completed by participants on a voluntary basis, some variables have more missing observations than observed observations. This is a challenging statistical problem because a household's decision to respond or not to a given question may be correlated with their household income, our choice variable of prediction. This means that normal regression methods will result in inconsistent estimators. For example, for a standard regression model $Y = X\beta$, we would have:

$$E(y_i|x_i) = x_i\beta + E(y_i|x_j = NA) \tag{1}$$

where x_j represents the variables $j = \{j_1, j_2, ..., j_k\}$ with x_j not reported. Put intuitively, the second term $E(y_i|x_j = NA)$ descibes how we would alter our prediction given that we have NA in columns j. It is possible that wealthier households are more likely to not self-report capital gains, and therefore, we should raise our expected probability of a household having annual income greater than $50,000 if we observe a missing value.(Another problem all-together is that households could have purposefully mis-reported-a complexity that we avoid all-together to simplify our analysis). The KNN method (described later) seeks to alleviate, but does not solve this problem. Perhaps due to this interaction between household income and missing data, several variables that intuitively would be a very good measure of household income, like wage/hr, weeks worked per year, and capital gains, are not only missing in many observations, but are not good predictors of household income.

One of the most pertinant characteristic of the data set is that the "over $50K" variable only takes on a value of true for 6.2% of the observations. As we noted in our original proposal, this means that a majority classifiers–predicting that income is under 50K for all observations, would have been "93.8% accurate". However, this is not a useful predictor, since we do not know if a set of new observations would be reprentative at all of our current data. In addition, it's important to consider weighting the relative importance of assigning a false positive vs. a false negative: for a business/marketing application of the data, it may be more important to predict all of the observations with income $>$ $50K, than correctly predict observation under $50K. In this sense, the majority classifier is undesireable, asit predicts no observations with income $>$ $50K, regardless of information.

3 Logistic Regression

The task of predicting whether a household self reports that its income is over or under $50,000 is a binary classification, as our outcome takes a value of either true (income over $50K) or false (income under $50K). We decided early on in the project to use logistic regression, and sought to find the best classification method within the subset of logistic regression classifiers.

3.1 Explanation of Logistic Regression model

Logistic regression transforms the response based on the predictors to

$$\pi(X) = \frac{e^{(X\beta)}}{1 + e^{(X\beta)}} \tag{2}$$

a transformation which looks like:

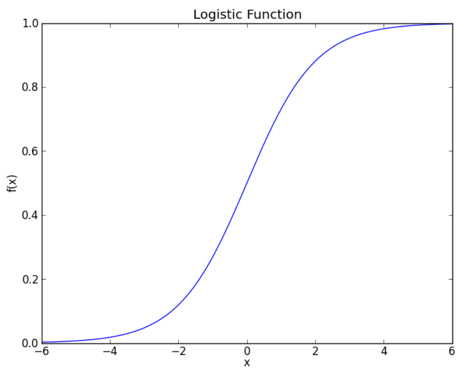

Figure 1: A graph of the logistic function for reference, graphed by the authors using the numpy module of python

3.2 Intuition behind logistic regression

This function can be viewed as a CDF (cumulative density function) and closely mirrors that of a normal distribution, and projects the space of $X\beta$ to the interval $[0, 1]$, allowing us to view the result as the probability that our outcome is 1. Our reasons for selecting a logistic regression model are simple: although it is more complicated than a basic regression mode $Y = X\beta$, logistic regression allows us to predict a binary outcome, while resulting in more interpretable coefficients β than "black box" methods such as neural networks or random forest.

4 Initial analysis

Our inital analysis was done in R, and much work was required to find and adapt the necessary statistical methods for use in python. We used the Akaike Information Criteria (AIC), defined as $AIC = 2k - 2ln(L)$, where k is the number of the parameters in the model, and L is the maximized value of the likelihood function - in our case, the logistical regression equation defined above. By maximizing AIC, we produced a subset of predictors that AIC evaluated as the "best" predictor sets. We noticed that among single-variable predictive methods, the education variable was one of the most effective, with a false positive error rate of 5% and false negative error rate of .7%. However, these rates are not accurate measures of performance, because we are testing our predictor with the same data we used to train. (Correcting our evaluation of model effectiveness will be discussed in the cross validation section).

5 Filling in missing observations with KNN

Andy spits his magic here.

6 Results: Model Selection

Andy also spits magic here

7 Cross Validation

An important aspect of our project is that our selected models predict accurately, not only for the sample for which we have data, but for the theoretical population. In this case, our population is less theoretical-it is the U.S. population represented by our census data. Because our prediction models are built by minimizing the error within our sample, it's unreasonable to assume that the model will have as low rates of error as those that are achieved by predicting the data used to generate the model.

This problem is usually addressed by the use of training sets and test sets. The data is initally partition, often around 3/4 for a training set and 1/4 for a test set. The training set is used to develop a prediction model, and the resulting model is used on the test set to obtain a theoretically unbiased estimate of model accuracy. The major downside to this approach is that valueable data points are lost when observations are assigned to the test sets, because these observations are no longer used to train the model. In situations were predictive power is important we want to minimize the loss of information that results from assigning data points to the test set.

This is the motivation for Cross validation

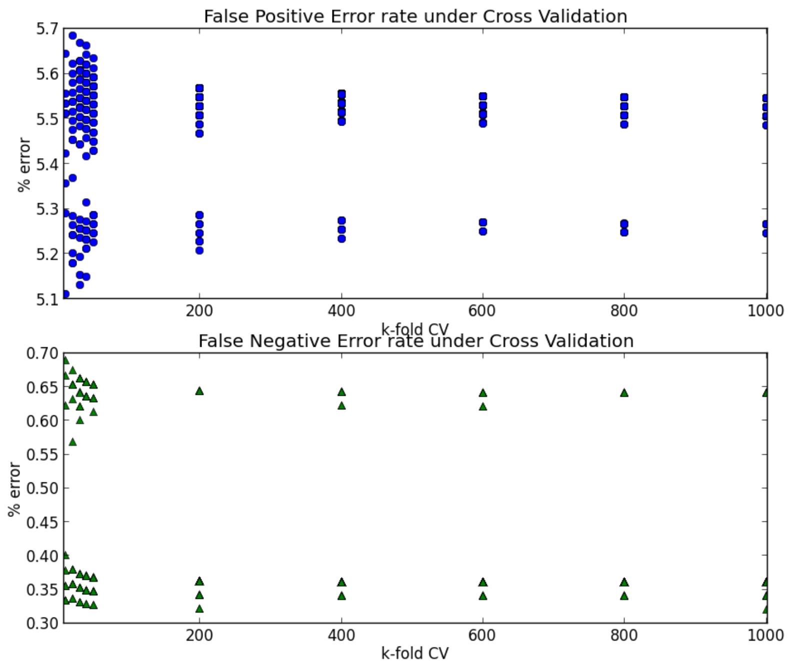

Figure 2: Error rate of education-only model, performed on 1,000 randomly selected observations

8 appendix

For a list of the variables in the data set, see