CS289 Initial Report:
Methods of handling missing data for classification

Rafael Valle[†] Jason Poulos[‡]

November 4, 2015

1 Motivation

Methods of handling missing data for neural networks classification model

Given that we plan to use NNets for income prediction ($income \geq \$50K/yr$) on the Adult dataset, we must handle missing data. This is less problematic for ML models such as random forest, decision trees, etc.

Item nonresponse is a common problem in survey data in several domains. Several techniques for data imputation (replace missing values with plausible ones) and direct estimation(all missing data is analyzed using a maximum likelihood approach) have been developed De Leeuw et al. [2003].

Proper statistical adjustement of missing data is very important, as naive solutions might introduce bias. We're interested in using the data to train neuronal networks. This must be taken into account becouse higher input values will result in a higher activation input.

In this project, we plan to evaluate different data imputation and direct estimation techniques within the context of using a neural network income classifier on the ADULT dataset. We plan to compare our results to previous techniques and models, such as bla bla bla, that addressed this same question.

2 Data

We plan to experiment with the Adult dataset from the UCI Machine Learning Repository [Lichman, 2013]. The dataset has 48,842 samples (train = $32,561$ and test = $1,6281$), and

[†] rafaelvalle@berkeley.com.

[‡] poulos@berkeley.edu.

3,620 (7.4%) of these samples contain missing values. The dataset contains 14 features: 6 continuous and 8 categorical. The prediction task is to determine whether a person makes over $50,000 a year; 24% of individuals in the training data make more than this amount.

Table 1 shows the test error rates obtained by the data set donor [Kohavi, 1996]. All error rates were obtained after removing samples with missing values. The error rate to beat is 14.05%.

Given the results of our experiments and if time permits, we may move to a much larger dataset, such as the 1940 full–count U.S. Census file [NAPP, Ruggles et al.]. The 1940 Census has about 100 million samples and 100 features.

Algorithm	**Error**
1 C4.5	15.54
2 C4.5-auto	14.46
3 C4.5 rules	14.94
4 Voted ID3 (0.6)	15.64
5 Voted ID3 (0.8)	16.47
6 T2	16.84
7 1R	19.54
8 NBTree	14.10
9 CN2	16.00
10 HOODG	14.82
11 FSS Naive Bayes	14.05
12 IDTM (Decision table)	14.46
13 Naive-Bayes	16.12
14 Nearest-neighbor (1)	21.42

Table 1: Test set error rates on Adult dataset for various algorithms, obtained after removal of samples with missing values and using the original train/test split. Source: Lichman [2013].

2.1 Patterns of missing values

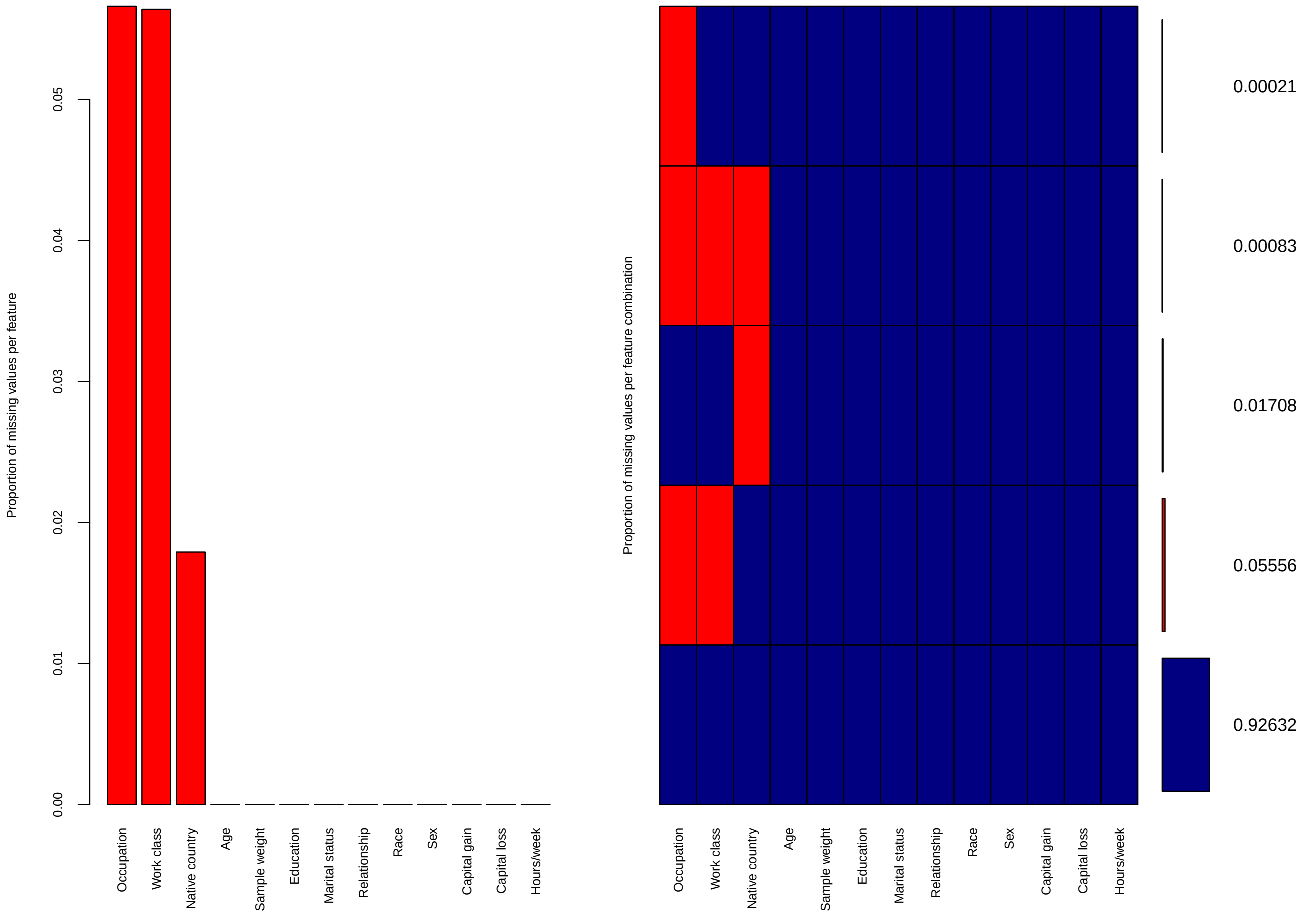

Figure 1: Histogram of proportion of missing values in each feature (Left) of Adult training set and aggregation plot of all existing combinations of missing and non-missing values in the samples (Right).

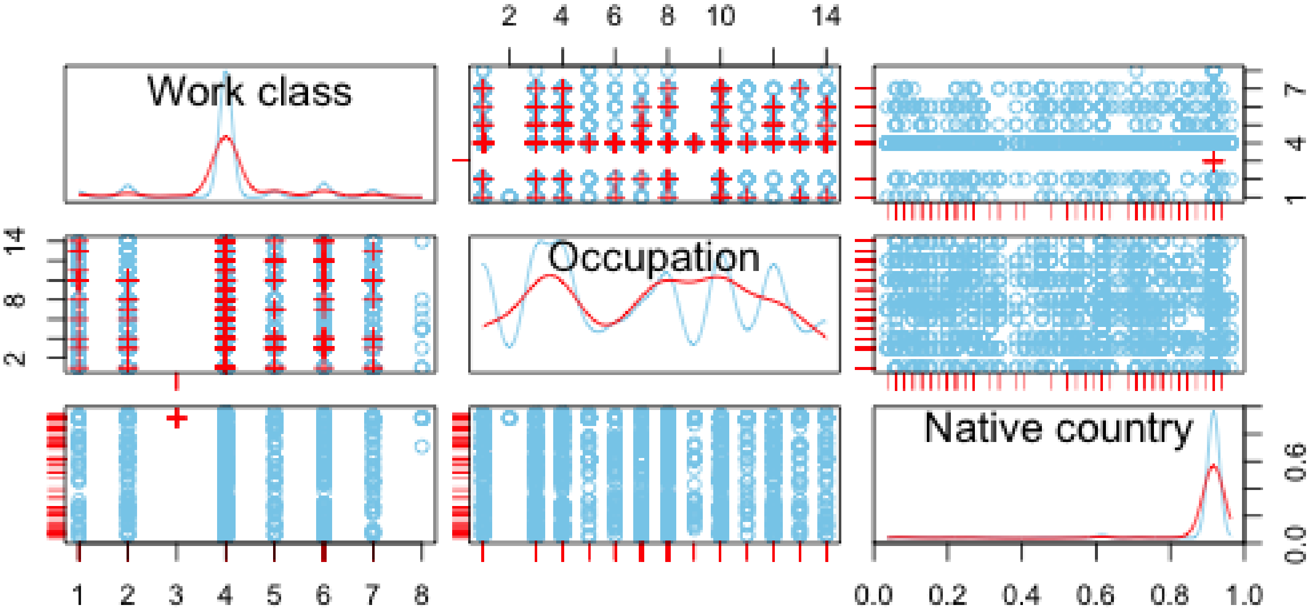

Figure 2: Scatterplot Matrix of features with missing values in Adult training set. Blue represents observed values and red represents missing values.

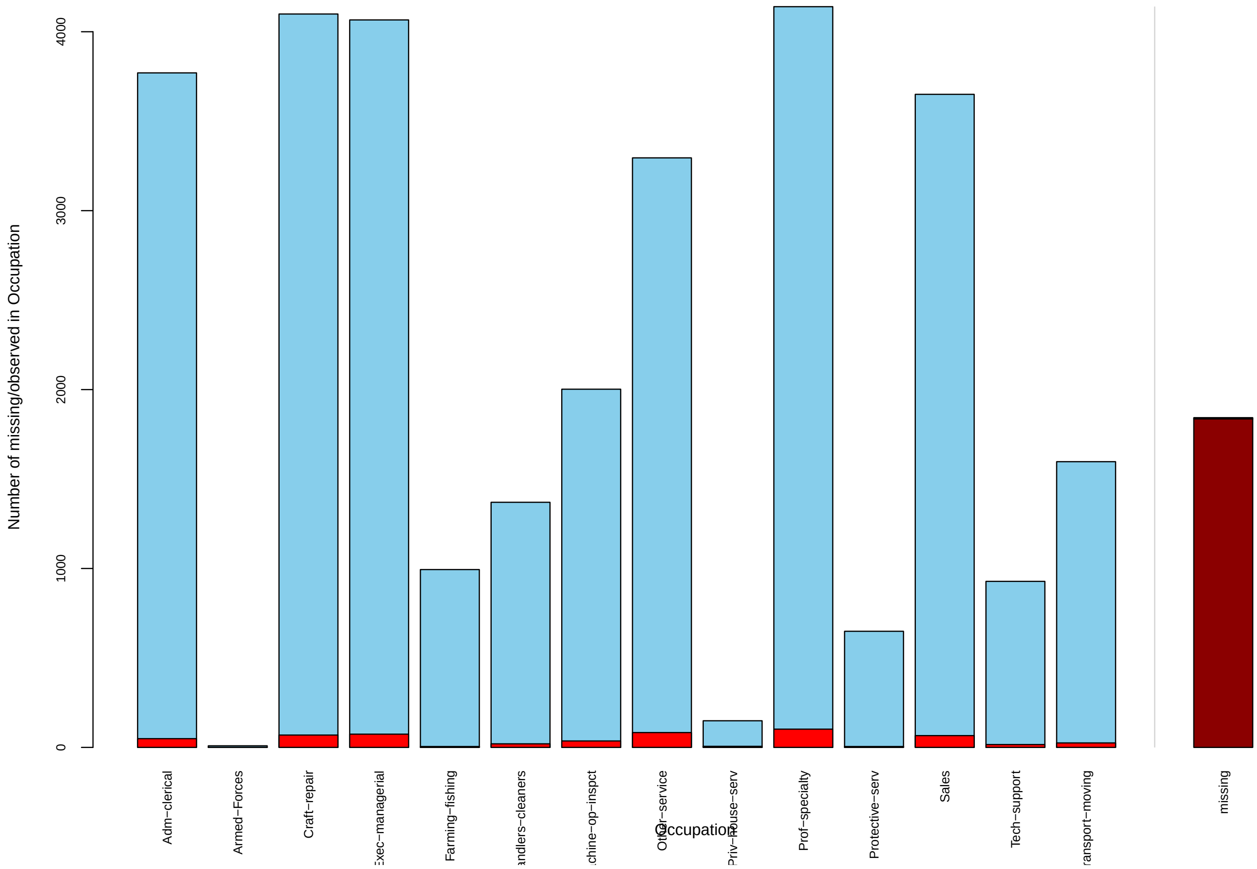

Figure 3: Barplot of the feature *Occupation* in the Adult training set (blue). Imputed values in *Work class* or *Native country* are highlighted in red.

3 Methods

3.1 Techniques for handling missing data

Assuming that the techniques below are easy to implement, we would like to compare their efficiency in imputing the missing values.

1. Basic Statistics : Replace the missing data with the mean or median of the feature vector. This is the most naive approach and since the missing variables in the ADULT dataset are all categorical, using the mean is not appropriated.

2. One-hot : Create an binary variable to indicate whether or not a specific feature is missing. This technique was mentioned by Isabelle

3. Nearest Neighbor Imputation : Recursively compute the K-Nearest Neighbors of the observation with missing data and assign median of the K-neighbors to the missing data. This technique is used in airbnb's fraud detection algorithm and explained in their website.

4. Logistic Regression : train a logistic regression model with all features except the feature with the missing variable to predict the missing value.

5. Bagging : Use one bag tree model for each predictor based on all other predictors.

6. Factor analysis : Perform some sort of factorization on the design matrix, project the design matrix onto the first two eigen vectors and replace the missing values by the values that might be given by the projected design matrix.

7. Find other features with distribution similar to the feature containing missing data and use this information (e.g. correlation) to fill in in the missing data. However, if two features are highly correlated, it might be better to remove one of them.

3.2 Neural networks for classification

References

Edith D De Leeuw, Joop Hox, Mark Huisman, et al. Prevention and treatment of item nonresponse. *JOURNAL OF OFFICIAL STATISTICS-STOCKHOLM-*, 19(2):153–176, 2003.

Ron Kohavi. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *KDD*, pages 202–207. Citeseer, 1996.

M. Lichman. UCI machine learning repository, 2013. URL `http://archive.ics.uci.edu/ml`.

NAPP. Minnesota population center. north atlantic population project: Complete count microdata. version 2.0 [machine-readable database]. *Minneapolis, MN: Minnesota Population Center, available at* ***https://www.nappdata.org****, year=2008.*

S. Ruggles, T. Alexander, K. Genadek, R. Goeken, M. Schroeder, and M. Sobek. Integrated public use microdata series (ipums): Version 5.0 [machine-readable database]. *University of Minnesota, Minneapolis, available at* ***http://usa.ipums.org****, year=2010.*