

# MISSING DATA IMPUTATION FOR NEURAL NETWORKS

Jason Poulos and Rafael Valle

*University of California, Berkeley*

*Abstract:* Researchers analyzing survey data typically choose decision trees or random forests for prediction tasks, largely because missing data and categorical variables are not easy to handle with neural networks. This paper investigates techniques for handling missing categorical data such that it is appropriate to neural network classifiers. We experiment on two benchmark datasets with missing categorical data, comparing a four-layer neural network with decision tree and random forests classifiers trained on differently imputed data and various levels of perturbation. We beat the state-of-the-art test error on both datasets using a random forests classifier, either trained on data imputed with logistic regression or data with one-hot encoding. We conclude from the results that the performance of the classifiers and imputation strategies generally depend on the nature and proportion of missing data.

*Key words and phrases:* missing data, imputation methods, neural networks, random forests

## 1. Introduction

Missing data is a common problem in survey data in various domains.

Random Forests and other decision tree methods are the method of choice for survey data, largely because missing data are not easy to handle with neural networks. We investigate techniques for handling missing categorical data such that it is appropriate to neural network classifiers. We assess the performance of neural network classifiers in comparison with decision tree and random forests classifiers trained on differently imputed data and for various degrees of perturbation.

After briefly reviewing related works in Section 2 we experiment on benchmark datasets and compare our results with the state-of-the-art in Section 3. Finally, we draw conclusions in Section 4.

## **2. Related work**

### **2.1. Neural networks for classification with categorical and continuous features**

Common techniques for handling categorical data in neural networks include encoding the categorical values into numeric values or using binary encoding. These techniques, however, have some drawbacks including unnecessarily increasing model complexity or feature dimensionality and not preserving the similarity information embedded between categorical values (Hsu, 2006). More elaborate techniques include information theoretic measures (Wang et al., 2008), training separate output units for each of

the allowed combination of values of the categorical independent variables (Brouwer, 2002), and using distance hierarchies (Hsu, 2006).

In the case of categorical variables, which by definition have no direct representation or computation scheme of the distance between its values, decision trees can be useful because they do not require distance metrics. However, their training process is slow given a large enough dataset and they might not be suitable for problems where the decision boundary between classes described by a second-order polynomial, for example (Fayyad et al., 1996).

**2.2. Techniques for handling missing data** Several techniques for data imputation (i.e., replace missing values with plausible ones) and direct estimation (i.e., all missing data is analyzed using a maximum likelihood approach) have been proposed (De Leeuw et al., 2003). We compare seven different imputation techniques within the following five categories which are commonly found in the previous work on missing data imputation (Batista and Monard, 2003):

***k*-nearest-neighbors (*k*-NN)** Compute the *k*-NN of the observation with missing data and assign the mode of the *k*-neighbors to the missing data.

**Classification algorithm** Train a classification algorithm — such as logistic regression, random forests or support vector machines (SVMs) — to

classify missing data given existing data.

**Random replacement** One observation with missing data is randomly replaced with another non-sampled observation.

**Mode replacement** Replace the missing data with the mode of the corresponding feature vector of the training set.

**One-hot** Create a binary variable to indicate whether or not a specific feature is missing.

### 3. Experiments

#### 3.1. Benchmark data sets

We experiment on two benchmark datasets from the UCI Machine Learning Repository: the Adult dataset and Congressional Voting Records (CVRs) dataset (Lichman, 2013). The Adult dataset contains  $N = 48,842$  observations and 14 features (6 continuous and 8 categorical). Missing values in this dataset are unknown survey responses. The prediction task is to determine whether a person makes over \$50,000 a year. The CVRs dataset contains  $N = 435$  observations, each the voting record of a member of the 98<sup>th</sup> U.S. House of Representatives for 16 key roll call votes identified by the Congressional Quarterly Almanac. Thus, the dataset contains 16 categorical features with three possible values: “yea”, “nay”, and missing. Missing

values in this dataset are not simply unknown, but represents values other up-or-down votes, such as voted present, voted present to avoid conflict of interest, and did not vote or otherwise make a position known. The prediction task is to classify party affiliation (Republican or Democrat).

We randomly split each dataset 2/3 for training and 1/3 for testing. The state-of-the-art for the Adult dataset is a Naive Bayes classifier that achieves a 14.05% generalization error after removing samples with missing values (Kohavi, 1996). The CVRs dataset donor claims to achieve a 90-95% accuracy using an incremental decision tree algorithm called STAGGER, although it is not known what train-test split is used or how missing values are handled (Schlimmer, 1987; Schlimmer and Granger Jr, 1986).

### 3.2. Patterns of missing values

Uncovering patterns of missing values in the dataset will help select strategies for imputing missing values. Figure 1 analyzes patterns of missing data in the Adult dataset, in which 7.4% of the observations contain missing values. Missing values occur in three of the categorical features: *Work class*, *Occupation*, and *Native country*. Since these the former two features are related, it is unlikely that these values are missing completely at random (MCAR); it is more likely, and much less desirable that the values are not missing at random (MNAR). The histogram (left) in Figure 2 shows *Work*

*class* and *Occupation* each have about 5.6% of missing values, and *Native country* has about 1.7% missing values. The aggregation plot (right) shows 5.5% of samples are missing values for both *Work class* and *Occupation*. Less than 2% of samples are missing just *Native country* and less than 1% are missing all three features.

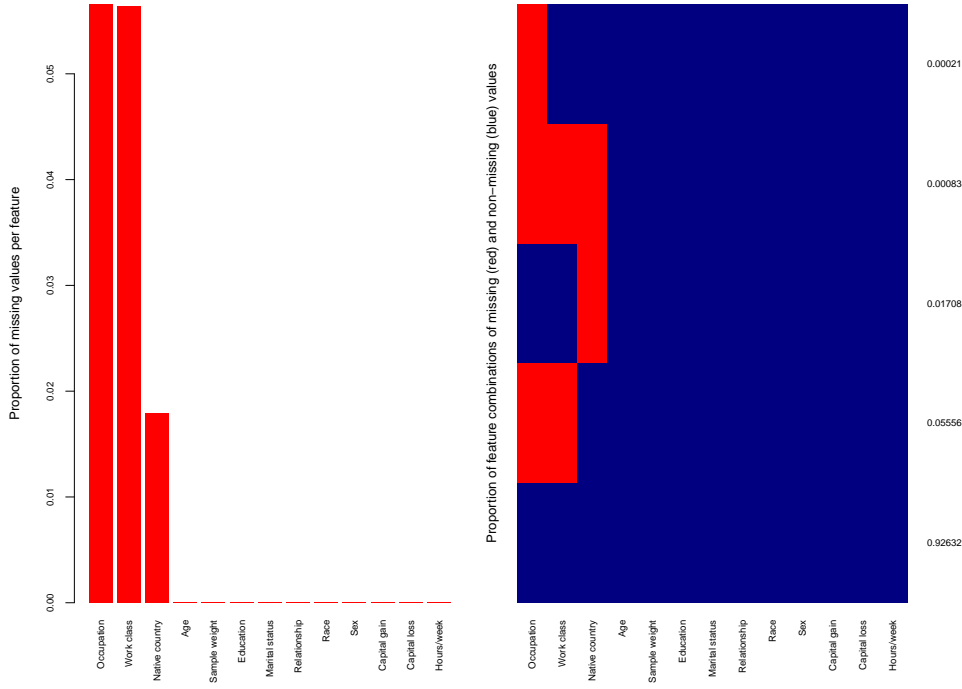


Figure 1: Histogram of proportion of missing values in each feature (Left) of Adult training set and aggregation plot of all existing combinations of missing and non-missing values in the samples (Right).

Figure 2 shows that 46% of the CVRs data contains missing values and all features contain missing data. About a quarter of missing data is in South Africa, which was a controversial amendment to amend the Export

Administration Act to bar U.S. exports to South Africa’s apartheid regime. Twelve percent of missing data is in the feature **Water**, which is a water projects authorizations bill, and 7% of missing data rests in the feature **Exports**, which is a tariff bill.

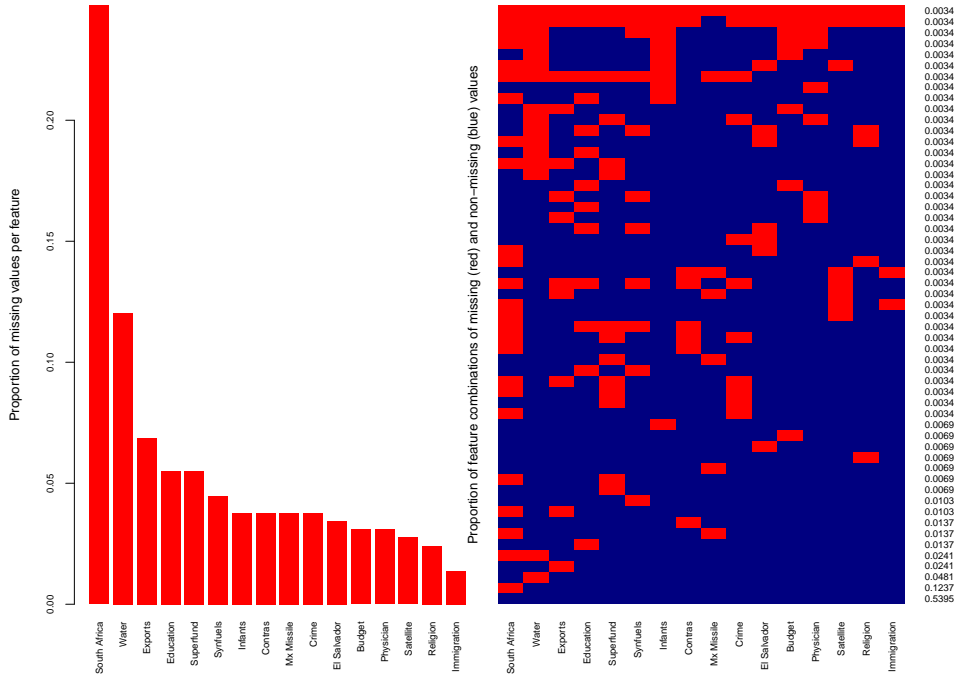


Figure 2: Histogram of proportion of missing values in each feature (Left) of CVRs training set and aggregation plot of all existing combinations of missing and non-missing values in the samples (Right).

### 3.3. Preprocessing

The first step in preprocessing the training data is artificially increasing the number of missing values in order to study the effect of larger amounts of missing data. We accomplish this by perturbing the categorical training

features by substituting nonmissing values with missing values so that each feature has a minimum missing data ratio, from 0-40%. For example if there 10 categorical features and 10 observations, a missing data ratio of 0.1 will perturb 10 values. Second, we implement one-hot encoding (-1,1) for the categorical variables. Third, we implement the imputation technique. Lastly, we standardize continuous features by subtracting the mean and dividing by the standard deviation of the feature. We preprocess the test data in the same manner, with the exception that there is no perturbation of test features.

### 3.4. Model training and hyperparameter selection

We train a four-layer neural network, with each of the two hidden layers having 1024 nodes. We use the adaptive learning rate method Adadelata (Zeiler, 2012) for the update rule. We explore the following hyperparameter space via Bayesian optimization (Snoek et al., 2012):

- Momentum schedule [0 to 1]
- Dropout regularization [No, Yes]
- Learning rate: [0.000001 to 0.01].

The goal of Bayesian optimization is to choose a point in the hy-

---

When imputing the missing data with mode replacement, we use the training set mode. We also use the training set mean and standard deviation to standardize test set features.



hyperparameter space that appropriately balances information gain and exploitation. Figure 3 shows the exploration of hyperparameter space during Bayesian optimization for both Adult and CVRs datasets. Each circle represents a candidate neural network classifier trained on a differently imputed and perturbed dataset. More circles appear in the plot for CVRs simply due to the fact that the training set is smaller. We see that most of the candidate models use dropout and have an initial learning rate close to the maximum of 0.01. The plurality of candidate models appear to either have momentum (1) or not (0). The candidate model with the lowest training error is selected for predicting on the test set.

### 3.5. Model assessment

We assess the performance of the neural network classifier in terms of test set error rate in comparison with decision tree and random forests classifiers on differently imputed data and for various degrees of perturbation. We use one-hot encoding to represent missing data when no imputation method is used. The results on the Adult dataset and CVRs dataset are plotted in Figures 4 and 5, respectively. For the Adult dataset, the random forests classifier trained on data imputed with logistic regression yields the lowest generalization error (13.85% error), beating the state-of-the-art by 0.2%. In comparison, random forests trained on data with no missing

data imputation matches the state-of-the-art (14.05%) and neural network trained on data imputed by random replacement performs considerably worse (14.37%). For the CVRs dataset, random forests trained on one-hot encoded data with up to 20% of the data perturbed beats the state-of-the-art by over 2% (2.77%). Neural networks trained on PCA-imputed data or no imputation also beat the state-of-the-art, with error rates of 3.47% and 4.16%, respectively.

Overall, the performance of the classifiers and imputation strategies depend on the dataset and amount of missing data. For the Adult dataset, random forests classifiers trained on data imputed with other classifiers (i.e., logistic regression, random forests, and SVMs) outperform other classifiers and imputation methods across different ratios of perturbed data. All classifiers trained on one-hot encoded data perform very poorly when the Adult dataset is perturbed. This is likely due to the fact that in the original, non-perturbed Adult dataset, missing values are concentrated in three features which may not be consequential for the prediction task. Perturbation exposes features that are more consequential to the prediction task to missing data.

Each of the three classifiers trained on one-hot encoded data perform well on the CVRs dataset. In this dataset, missing values represent poten-

tially valuable information for the prediction task and can be more useful for the classifier than the imputed value for certain features. This is why it is not implausible that a classifier trained on perturbed, one-hot encoded data can have lower generalization error than a classifier trained on non-perturbed, one-hot encoded data.

#### 4. Conclusion

Neural networks have become a popular machine learning algorithm in many domains, in part due to the ability of neural networks to “learn” how to engineer features. However, researchers analyzing survey data typically choose decision trees or random forests for prediction tasks because missing data and categorical variables are not easy to handle with neural networks. This paper investigates techniques for handling missing data for training neural network classifiers.

We compare the predictive performance of a four-layer neural network against decision tree and random forest classifiers trained on datasets with differently imputed data. We assess performance in terms of test set error for different levels of perturbed training data, from 0% (no perturbation) to 40% perturbation. We beat the state-of-the-art test error on the Adult dataset by 0.2% using a random forests classifier trained on data imputed with logistic regression. Random forests trained on perturbed and one-hot

encoded data outperforms the state-of-the-art on the CVRs dataset by over 2%.

We conclude from the results that the performance of the classifiers and imputation strategies generally depend on the nature and proportion of missing data. For the Adult dataset, random forests classifiers trained on data imputed with other classifiers outperform other classifiers and imputation methods across different ratios of perturbed data, while classifiers trained on one-hot encoded data perform very poorly on perturbed training data. This finding can be explained by the fact that missing values in the Adult dataset are concentrated in three features which may not be consequential for the prediction task, and perturbation exposes features that are more consequential to the prediction task to missing data. For the CVRs dataset, each of the three classifiers trained on one-hot encoded data perform well across different levels of perturbation. This finding can be explained by the fact that missing values represent potentially valuable information for the prediction task in the CVRs data.

For future work, we will further explore the idea that missing data may have more predictive value than imputed data in certain domains. A related question is whether neural networks can outperform decision trees or random forests trained on one-hot encoded data by learning different

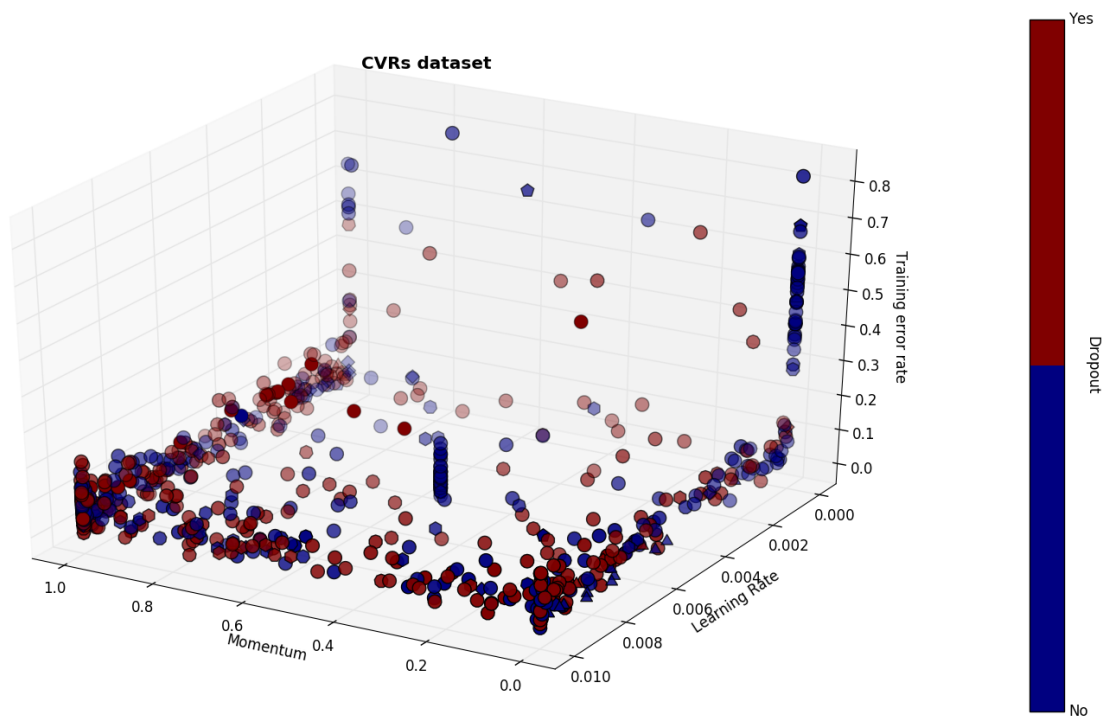
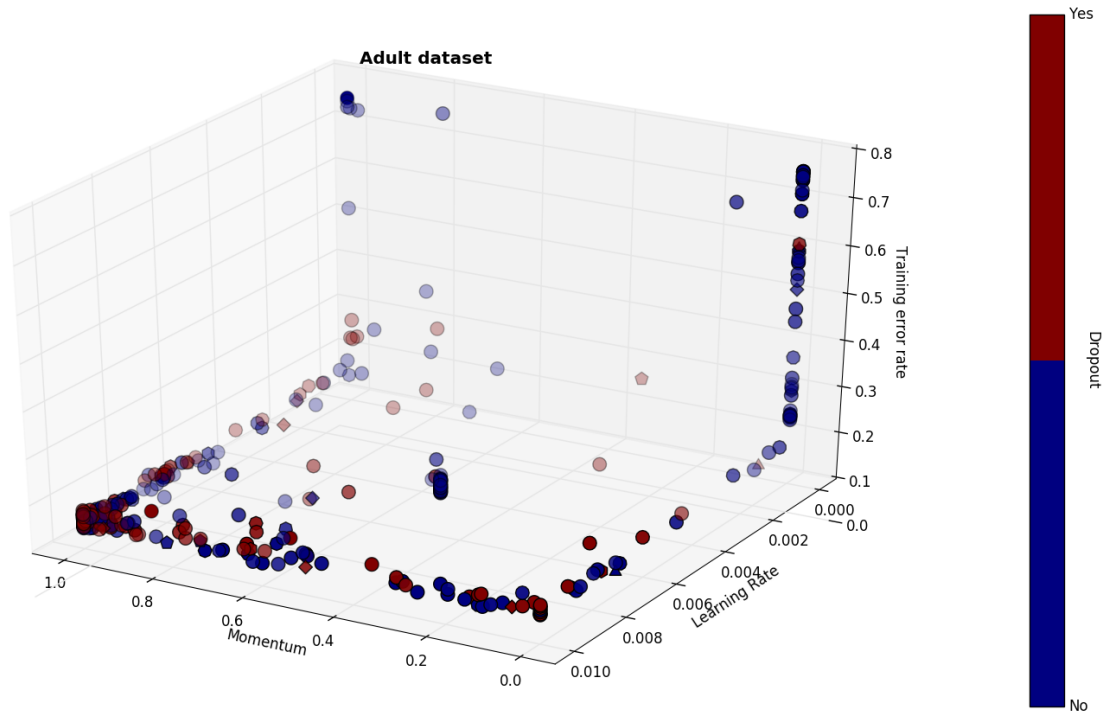
types of missing values. For instance, can neural networks engineer features that reflect the different types of missing values in the CVRs data, where missing values represent any action other than an up-or-down vote (e.g., voted present)? Answers to these questions will help guide researchers in choosing which imputation strategies and classifiers to use for prediction problems in different domains.

### **Supplementary Materials**

The code used for this project is available on Github (<https://github.com/rafaelvalle/MDI>).

### **Acknowledgements**

We thank Isabelle Guyon for advice and the idea for the paper. We also thank Joan Bruna and seminar participants at the University of California, Berkeley for comments. This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE 1106400. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.



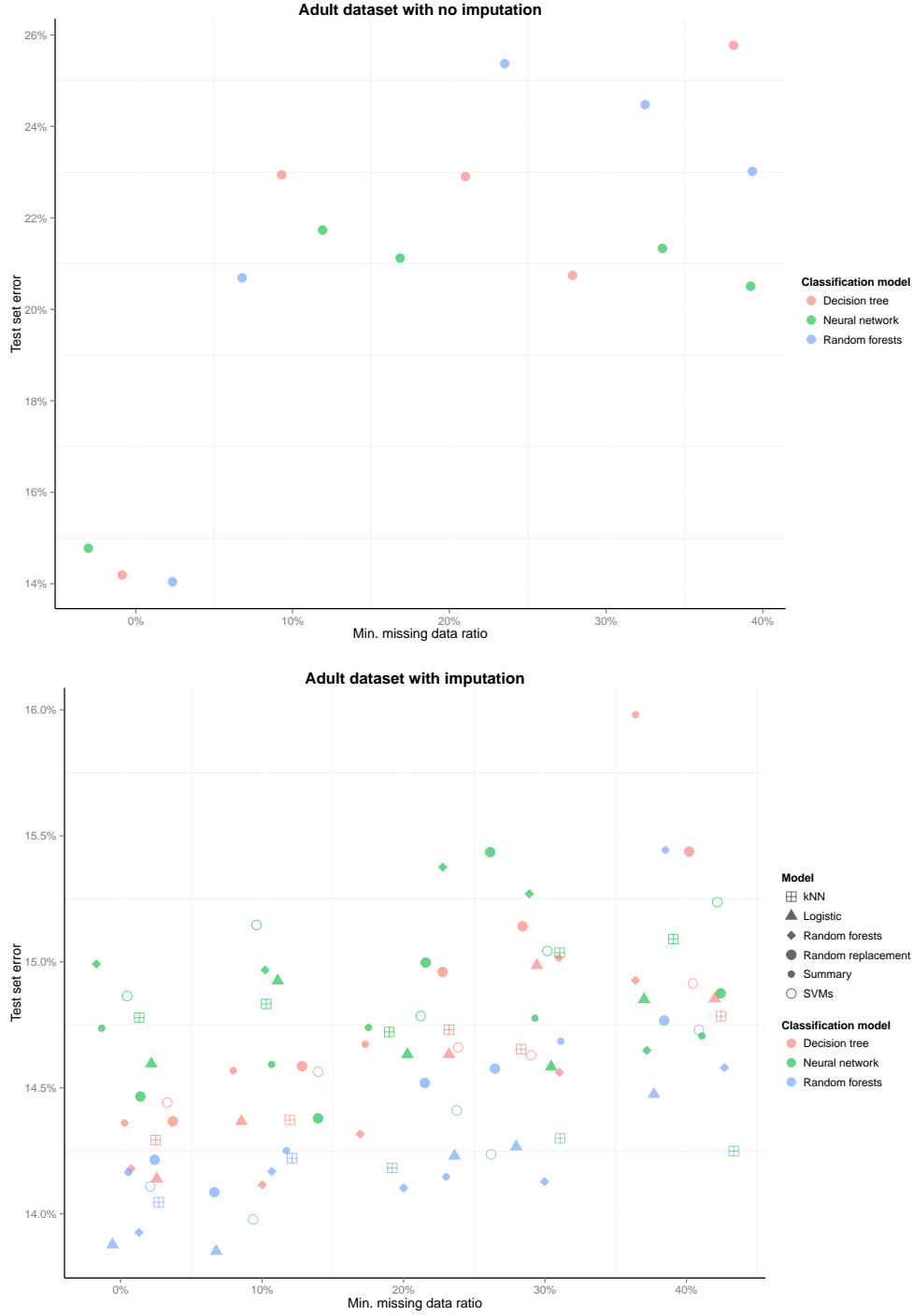


Figure 4: Error rates on the Adult test set with (bottom) and without (top) missing data imputation, for various levels of perturbed training features (x-axis). One-hot encoding is used to represent missing data in the absence of imputation. The decision tree and random forests classifiers are trained with maximum depths of 8 and 16, respectively.

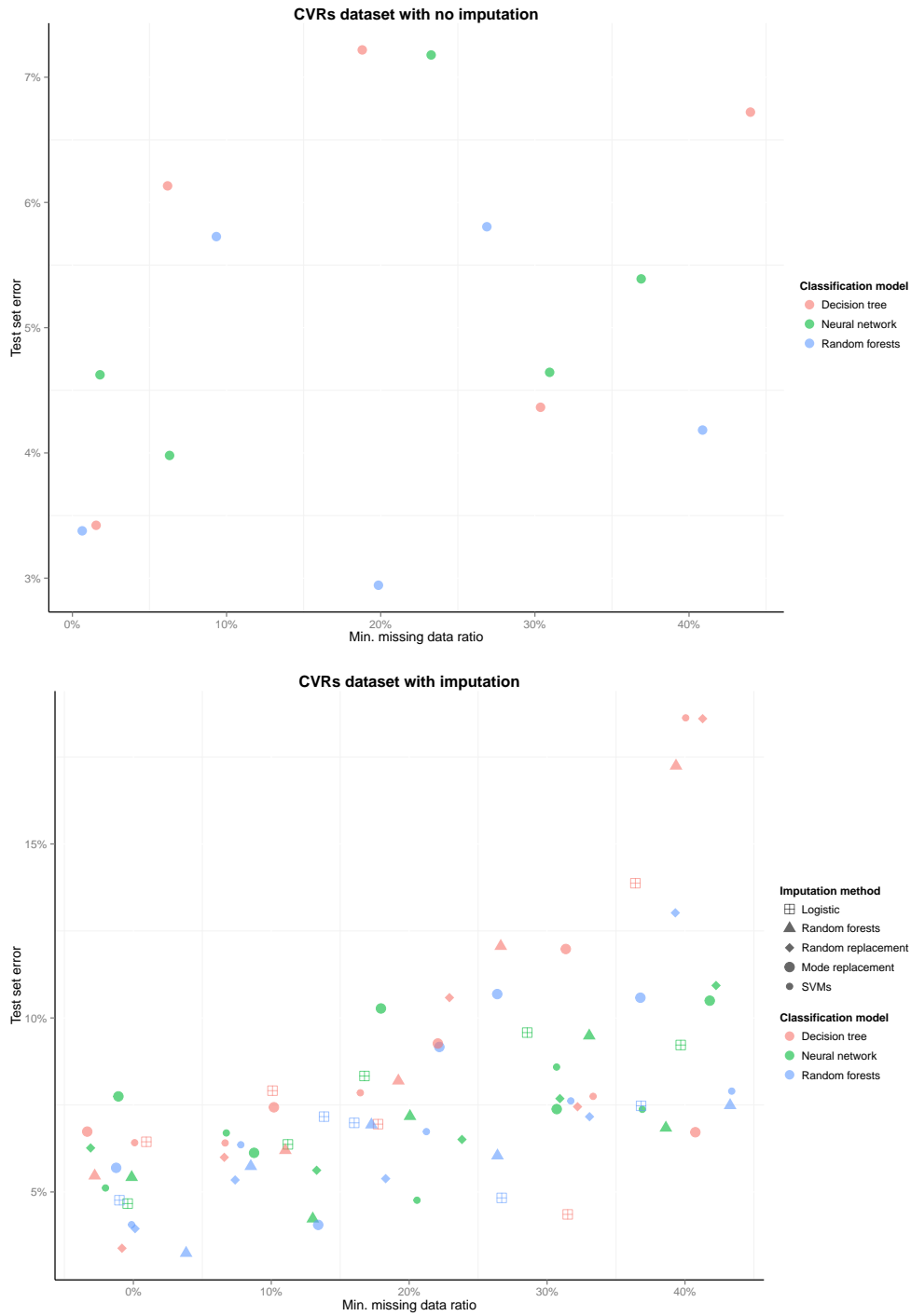


Figure 5: Error rates on the CVRs test set with (bottom) and without (top) missing data imputation.

See footnotes for Figure 4.



## References

- Batista, G. E. and Monard, M. C. (2003). An analysis of four missing data treatment methods for supervised learning. *Applied Artificial Intelligence*, 17(5-6):519–533.
- Brouwer, R. K. (2002). A feed-forward network for input that is both categorical and quantitative. *Neural Networks*, 15(7):881–890.
- De Leeuw, E. D., Hox, J., Huisman, M., et al. (2003). Prevention and treatment of item nonresponse. *Journal of Official Statistics-Stockholm*, 19(2):153–176.
- Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37.
- Hsu, C.-C. (2006). Generalizing self-organizing map for categorical data. *Neural Networks, IEEE Transactions on*, 17(2):294–304.
- Kohavi, R. (1996). Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *KDD*, pages 202–207. Citeseer.
- Lichman, M. (2013). UCI machine learning repository.
- Schlimmer, J. C. (1987). Concept acquisition through representational adjustment.
- Schlimmer, J. C. and Granger Jr, R. H. (1986). Incremental learning from noisy data. *Machine learning*, 1(3):317–354.
- Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–

2959.

Wang, H., Xing, G., and Chen, K. (2008). Categorical data transformation methods for neural networks. In *IKE*, pages 262–266.

Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Department of Political Science, University of California, Berkeley, CA 94720-1950

E-mail: `poulos@berkeley.edu`

Center for New Music and Audio Technologies, University of California, Berkeley, CA 94720

E-mail: `rafaelvalle@berkeley.com`