Working Paper.

UNITED NATIONS
ECONOMIC COMMISSION FOR EUROPE

CONFERENCE OF EUROPEAN STATISTICIANS

Workshop on Statistical Data Editing
(Neuchâtel, Switzerland, 18-20 September 2018)

Two-phase and double machine learning for data editing and imputation

Prepared by Susie Jentoft and Li-Chun Zhang, Statistics Norway

Abstract

Machine learning (ML) is becoming increasingly popular and accessible with the increased capabilities of modern computing. We investigate the use of supervised ML in data editing and extend it to two-phase and double ML approaches. As an example of two-phase ML, a random forest (RF) is grown at phase-1 for classification (or imputation), and another RF is grown at phase-2 to classify when the phase-1 RF classification align with new observations. Whereas for double ML, at stage-1 several RFs are grown for different variables separately; the prediction results from stage-1 are then combined into a single dataset, on which a stage-2 RF is grown to learn the circumstances under which the stage-1 RFs perform best. We use data from the Norwegian labour force survey to explore and illustrate these concepts. The approaches presented provide some directions towards greater automization of data editing and imputation processes. Their applications could potentially both save resources and improve quality. Some topics are identified for future research and development.

I. Introduction

1. Machine learning (ML) is gaining interest with the increase of machine power and our understanding on how we can apply these techniques. The question is now not *if* we can use them but *when* it is appropriate to use them in official statistics. This paper explores and illustrates ideas of what we call two-phase and double machine learning in the context of data editing.

2. Decision trees are a supervised ML approach to predict a target variable based on observed training data. Single decision trees, by themselves, can be visually appealing and easy to use and understand. Additional theoretical developments such as pruning, bagging, boosting and random forests has also made decision trees competitive in predictive power compared to more traditional modelling. For statistical data editing decision trees can provide a method of imputation.

3. Random forests (RF) are a progression where many trees are grown using bootstrap, i.e. a sample with replacement from the original data (Breiman, 2001). This allows an out-of-bag error rate to be calculated, which is the error in the classification of the unsampled observations for each bootstrap. No pruning is done to the trees so there is usually an overfitting of the data. This equates to low bias but high variance. In addition, at every split, a random selection of the predictive variables is taken. The size of the selection is usually $\sqrt{M}$, where M is the total number of predictive variables to select from. This decreases the correlation between trees, hence the variance of the prediction.

II. Two-phase ML

A. Introduction

4. Under a two-phase ML approach, we first build a predictive algorithm e.g. for the imputation of a variable, followed by a second algorithm aiming to predict which units will be treated successfully. RFs are used in both stages to test this concept. Two-phase ML may be used in data editing where we want to

create an imputation procedure for values we believe are incorrect: the 2nd phase ML provides then a *conditional* measure of the uncertainty of phase-1 ML. It has also potential in data collection, where the 2nd phase ML determines e.g. whether imputation may be better than trying to collect data by an inferior instrument such as indirect interview, or a costly approach such as call-back.

5. For an example of two-phase ML, we study the variable *partial absence (from work)* in the Norwegian Labour Force Survey (LFS), which is a large, continuous, rotating panel survey of around 24 000 people per quarter. Previous studies have indicated quality issues of this variable among those individuals interviewed through proxy interviews, where the person's spouse or parent responds on their behalf (Jentoft, 2018). This has led to what is believed to be a systematic underreporting error of partial absence and imputation is recommended. For this illustration, we assume that the aim of the 2nd phase is to learn in which circumstances the imputation technique learnt in phase 1 produces imputed values that coincide with the proxy interviews.

B. Methods

6. In phase 1, we create a 2/3 training data set for learning the RF, including those who were employed in the LFS in 2016 based on direct interview. Direct interviews, those that are done directly with the person selected for the survey, are regarded as the "gold standard" here. We included 20 additional variables to aid in prediction, most come from administrative registers, but also other variables, for example if the reference week includes a public holiday.

7. The R package *randomForest* was used for implementing the Classification and Regression Tree (CART) algorithm to grow the trees in the random forest. It is based on the description of Breiman et al. (1984). A CART is a binary decision tree that is constructed by splitting a node into two child nodes repeatedly, beginning with the root node that contains the whole learning sample. Splitting criterion uses the Gini impurity index defined as

$$GiniGain(D,X) = Gini(D) - \frac{N_1}{N} Gini(D_1) - \frac{N_2}{N} Gini(D_2)$$

where X is a given feature, D is the node on which the split is to be made, D_1 and D_2 are the two child nodes created by splitting D, and N, N_1, and N_2 are the number of elements in the parent, and two child nodes respectively. Furthermore $Gini(N)$ is defined as

$$Gini(N) = 1 - \sum_{k=1}^{K} p_k^2$$

where K is the number of classes in the node and p_k is the fraction of items with class k.

8. Given that partial absence occurs relatively infrequently (about 13%) we have an imbalance in the outcome classes. This can be a problem for learning RF algorithms as the Gini impurity index aims to minimize the *overall* error rate, usually resulting in greater classification of the majority class given RFs majority vote rule (Chen, Liaw, & Breiman, 2004). In order to adjust for this, we test random over-sampling with replacement of the minority class in phase-1, yielding the levels 25% and 50% in our training dataset. This is a technique which is relatively simple to execute, has been shown to improve the performance of RFs, and is competitive against other more complex balancing techniques (Batista, 2004).

9. RFs were set to grow 100 trees. Visual inspection of standard measures indicated this was enough for out-of-bag error rates to stabilize. We used the standard majority vote rule however also compared the outcomes with a proportional vote RF. The latter is where for each tree grown the outcome is the majority vote from the terminal node but the result of the entire RF is the proportion of votes from all trees. Table 1 shows the possible outcomes in phase 1 (majority vote RF) when comparing the predicted class in the test dataset (1/3) against the observed value.

Table 1. Summary of terms of outcomes

		Observed condition	
		Negative	**Positive**
Predicted condition	**Negative**	True negative (TN)	False negative (FN)
	Positive	False positive (FP)	True positive (TP)

10. We use three standard measures to assess Phase 1:

$$True\ positive\ rate\ (sensitiviy) = \frac{TP}{(TP + FN)}$$

$$False\ discovery\ rate = \frac{FP}{(FP + TP)}$$

$$Accuracy = \frac{TP + TN}{TOTAL}$$

11. By definition, any performance outcome that uses values from both rows in

12. Table 1 will be affected by the distribution of the outcome class (accuracy and true positive rate), but the false discover rate is not affected by this. We also look at measures which are important for producing statistics:

$$Predicted\ prevalence = \frac{(FP + TP)}{TOTAL}$$

$$Balance = \frac{FP}{FN}$$

13. In terms of producing rate statistics from the outcome of the phase 1 (RF_1), if $FP = FN$ or the $balance = 1$ we should produce unbiased estimates.

14. In phase 2, we created a training dataset (2/3) of those proxy interviews. We ran RF_1 for the new training dataset, to create a new outcome variable, *aligned,* for the congruency between the RF_1 predicted values and the observed proxy value

$$align_i = \begin{cases} 1 & if\ pred_{i,1} = obs_{i,1} \\ 0 & if\ pred_{i,1} \neq obs_{i,1} \end{cases}$$

15. Variables included in RF_2 were similar to those used in phase 1 with the addition of the predicted outcome from RF_1, and the variables who gave the interview (spouse or parent) and time difference between reference week and interview date.

C. Results

(a) Phase 1

16. Results from the first phase show a low true positive rate when the observed proportion is used in the training data from the standard majority vote RF. As the proportion of occurrences is artificially increased in the training data, the true positive rate also increases, i.e. the algorithm predicts more positive occurrences of partial absence (Table 2). However, the false discovery rate also increases and the overall accuracy decreases.

17. When comparing the estimate for the mean rate of the target variable among the methods we see that the RF using the proportion of votes produces an estimate close to the observed estimate. For the RF using majority vote, it appears that increasing the proportion of occurrences to 0.5 in the training dataset produces an estimate relatively close to the observed rate.

Table 2. Results from RF_1

Training data (2/3 of direct)	True positive	False discovery	Accuracy	Balance FP/FN	Mean (vote)	Weighted mean (vote)	Mean (proportion)	Weighted mean (proportion)
Observed (test data)					0.13*	0.13*	0.13*	0.13*
Observed	0.20	0.48	0.86	0.23	0.05	0.05	0.14	0.13
25%	0.29	0.53	0.86	0.47	0.08	0.08	0.19	0.17
50%	0.40	0.67	0.81	1.39	0.16	0.16	0.32	0.31

*observed and not from RF

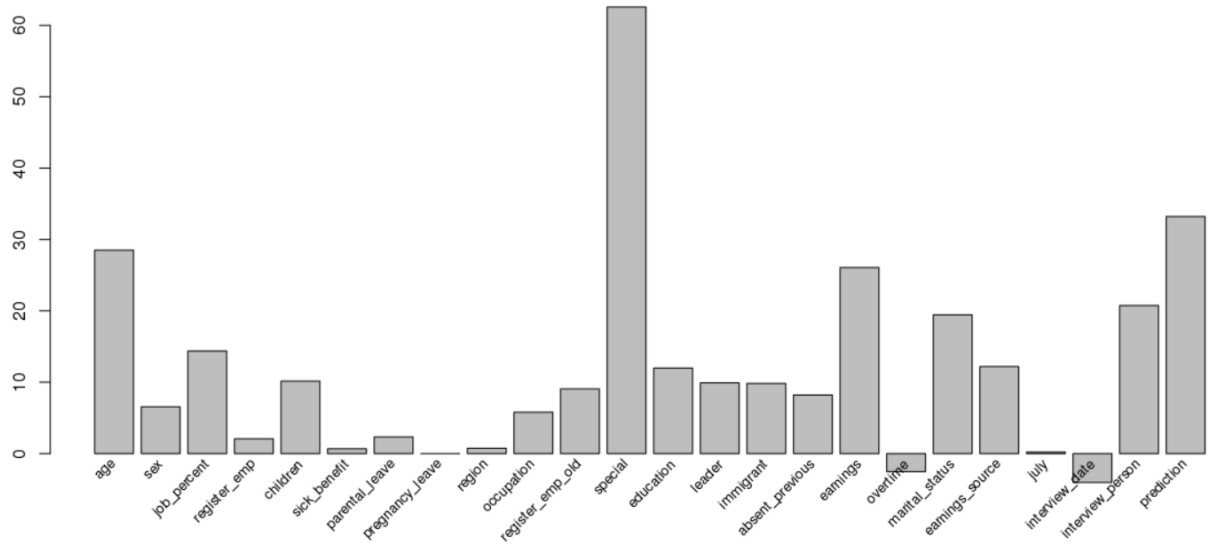
(b) Phase 2

Table 3. Results from phase 2 RF

Correct = positive	True Positive	False discovery	MSE	Accuracy
Observed %	0.9912	0.0872	0.0939	0.9048
25%	0.9896	0.0912	0.0984	0.9016
50%	0.9911	0.1355	0.1406	0.8456

18. The true positive and false discovery rates confirm that proxy interview yields a lower proportion of partial absence compared to direct interview. Variable importance in terms of its contribution to the RF is calculated by permuting the out-of-bag data (Liaw & Wiener, 2002). In **Error! Reference source not found.** we see that the greatest contribution for predicting alignment in phase 2 is the variable called *special* which indicates if the reference week includes a public holiday (also the strongest phase-1 predictor). The predicted outcome from phase 1 (*pred1*) and earnings (*loglonn*) contribute next most.

Figure 1. Variable importance for two-phase ML (phase 2)



C. Discussion

19. Based on our experience with the first phase of our two-phase ML, we see that the standard RF using a majority vote rule for summarising the forest is not able to predict the variable of interest (partial absence) unbiasedly when the training dataset is unadjusted. There appears to be a clear favouritism towards the most common class for this variable (no absence), which leads to serious under-estimation. Weighting did not contribute to reducing the difference between the observed and predicted results.

20. Increasing the prevalence of the rarer class in an artificial training dataset using replacement sampling increased the ability to detect true positive values, while increasing the false negative rate and decreasing the overall accuracy. We notice that there does not appear to be a theory regarding how such manipulations can achieve unbiased prediction.

21. Alternately, using the proportion of votes at the terminal nodes produced an estimate which was in agreement with the observed data. However, this approach does not result in class classification, so an additional step is needed if it is required to impute classes (for example a random draw according to the proportion or restricted hot-deck method).

II. Double ML

A. Introduction

22. Next we introduce the idea of double machine learning. In stage-1 we use the RF algorithm for several classification problems and at stage-2 a second RF is grown to learn *conditionally* the results of the stage-1 ML. Unlike the two-phase ML above, in double ML we use the *same* units in both stages.

B. Methods

23. Again, we use data from the Norwegian LFS 2016, this time investigating 5 binary variables of interest: *unemployment*, *employment*, *partial absence*, *overtime* and *temporary job*. In stage-1 RFs are learnt for these outcome variables separately using additional variables from administrative data. The training dataset uses 2/3 of all individuals in the LFS sample. In stage-2 we take the union of the stage-1 datasets and use additional standardized individual-level (feature) variables and dataset-level (contextual) variables. Feature variables include all variables used in stage-1, and a measure for the strength of the prediction ($pred\_good$) defined as:

$$pred\_good_j = abs(2 * pred_j - 1)$$

where $pred_j$ is the proportion-predicted value from an RF_i in stage-1 for unit j. This takes the maximum value 1 when $pred_j$ is 0 or 1, and the minimum value 0 when $pred_j$ is 0.5. An additional variable for the time difference between interview date and the reference week was included.

24. We include in stage-2 also dataset-level (contextual) variables: the occurrence of the variable of interest ($occurence$) in the training dataset, and a dataset identifier. We also use Tjur's pseudo R^2 (2009) as a measure of the explanatory power of the stage-1 RF, calculated from a full logistic regression model fitted to the training dataset. The contextual variables are shown in Table 4.

Table 4. Summary of a sample of variables in RF stage-2

	Occurrence (training data)	Pseudo R^2	Mean pred_good
Unemployment	0.03	0.2348	0.9645
Partial absence form work	0.10	0.1445	0.8313
Employment	0.73	0.8298	0.9325
Overtime	0.07	0.1032	0.8749
Temporary job	0.05	0.0972	0.8998

25. For stage-2 ML rhe combined dataset is divided into 2/3 training and 1/3 test datasets.

C. Results

(a) Stage 1

Table 5. Stage I results

	Accuracy	False discovery	True positive	Predicted occurrence	Observed occurrence (test data)
Unemployment	0.98	0.41	0.23	0.01	0.02
Partial absence form work	0.90	0.50	0.16	0.03	0.10
Employment	0.96	0.03	0.98	0.74	0.73
Overtime	0.93	0.42	0.05	0.01	0.07
Temporary job	0.95	0.34	0.07	<0.01	0.05

26. Table 5 gives a summary of the results from stage-1 RF. The variable *employment* had by far the highest true positive rates and lowest false discovery rates. From Table 4 we see that this variable had the most balanced occurrence (closest to 0.5 of the variables investigated) and the highest R^2. At the other end, the RF for predicting whether an individual had worked overtime had a very low true positive rate, and high false discovery rate. The predicted occurrence of the variable was also low compared to that observed in the test data. In addition this variable has a low R^2.

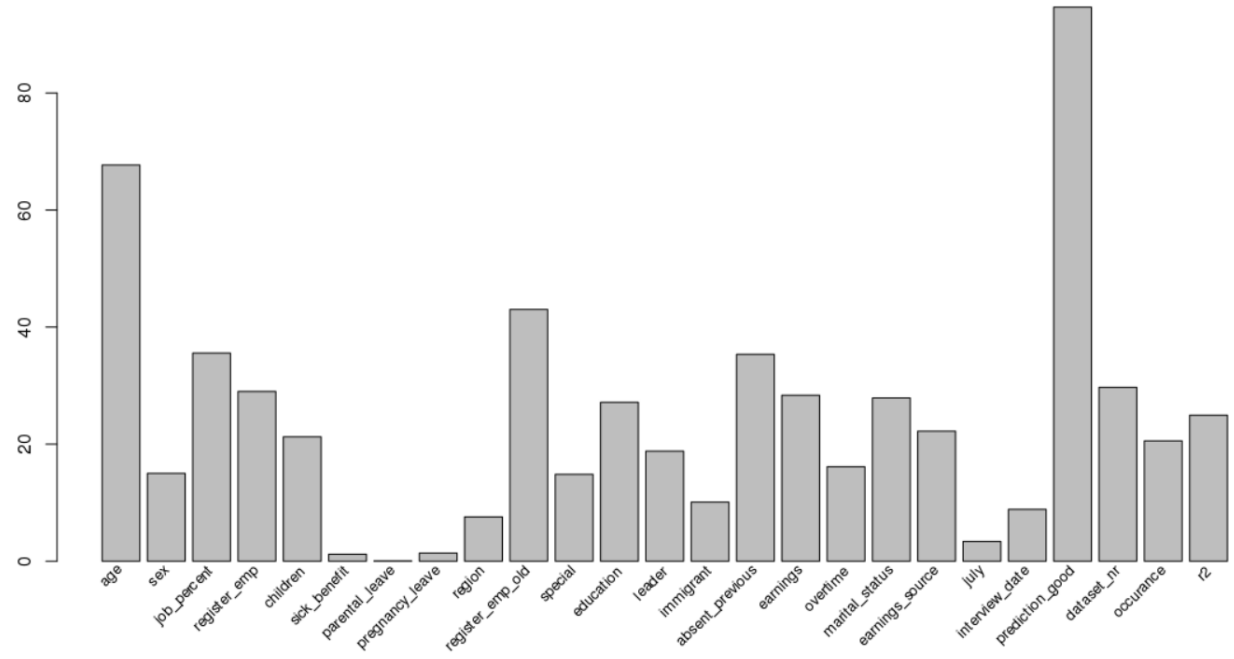
(b) Stage 2

27. The stage-2 RF attempts to learn and predict when a classification from stage-1 is good. Table 6 gives a summary showing that in general the prediction accuracy is high for all the variables we investigated. However, again there is an imbalance in the data and it appears the RF strongly favours the majority class, which in this case is that the class = correct. The true negative and false negative rates are shown and indicate that the algorithm is unable to identify when the predictions will be incorrect (which is the rarer event). This can be seen by the high predicted proportion of classification to class level correct which was 0.99 or more for all variables.

Table 6. Stage-2 results

	Accuracy	False negative FN/(FN+TP)	True negative TN/(TN+FP)	Predicted proportion classified correctly	Proportion classified correctly based on stage-1
Unemployment	0.98	<0.01	0.02	>0.99	0.98
Partial absence form work	0.98	0.01	0.07	0.99	0.90
Employment	0.99	<0.01	0.07	>0.99	0.96
Overtime	0.99	<0.01	0.02	>0.99	0.93
Temporary job	0.99	<0.01	0.03	>0.99	0.95

Figure 2. Variable importance from Stage-2



Variable importance, shown in

28. Figure 2, indicates which variables contribute most to the stage-2 RF, i.e. most useful in predicting whether stage-1 classification is correct. The variable *pred_good* appears to be the most important and predictive for stage 2. Age is also relatively important although this may be correlated with the variable of interest and reflect the class in which the individual belongs. We have seen that the ability for the RF to predict class level = 1 is not the same as class level = 0 and this may be reflected by some of the variables. Our chosen contextual variables do not appear particularly useful in this case.

D. Discussion

29. The double ML presented here gives an idea for how to approach the task of automizing imputation procedures. At this point, we have encountered a number of problems when using RFs for unbalanced class levels and this is an area for further investigation. We have identified a potentially important variable in determining the best circumstances for using an RF (majority vote) algorithm for imputation as when there are strongly polarized proportion votes (high *pred_good* values). This is sometimes referred to as low overlap at the terminal nodes and other studies have indicated this as a more important measure than balance (Batista, 2004).
30. A note should be made that some caution must be taken in interpreting variable importance results presented here. Studies have indicated that standard permutation variable importance measures may not perform accurately in cases of highly unbalanced class data and with the presence of numeric and categorical explanatory variables with many levels (Janitza & Carolin Strobl, 2013). Testing alternative variable importance measures such as area under the curve permutation measures is also an area for further study, although the alternatives may be much more computation-intensive.

III. Concluding remarks

31. In this paper, we have explored two ML approaches for determining conditionally in which context it may appropriate to apply an RF algorithm. In the two-phase ML, where the units in phase 1 and 2 were different, we learnt an algorithm to predict alignment. Provided this can be done accurately and quickly the technique has many applications, for example in predicting quality of sources for data collection, potentially reducing response burden if indirect interviews can be dropped in certain circumstances. It also has applications in data editing for the detection of incorrect values and the imputation of them.
32. Double ML aims at automizing the imputation process and learning the circumstances under which a predictive algorithm works well. This has potential resource- and time-saving benefits for official statistics organisations. While only one type of ML algorithm was explored in this study we see the possibility for testing other ML and building on this idea to become increasingly more powerful and effective at solving real-world data editing and imputation problems.

References

Batista, G. E. (2004, 6 1). A study of the behaviour of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, ss. 20-29.

Breiman, L. (2001). Random Forests. *Machine Learning*, ss. 5-32.

Chen, C., Liaw, A., & Breiman, L. (2004). *Using random forest to learn imbalanced data.* Berkley: University of California.

Jentoft, S. (2018). Mode effects from proxy interview in the Norwegian LFS. *Labour Force Survey Workshop 2018.* Reykjavik.

Liaw, A., & Wiener, M. (2002, 2 3). Classification and regression by randomForest. *R News*, ss. 18-22.

Tjur, T. (2009). Coefficients of determination in logistic regression models - A new proposal: The coefficient of discrimination. *American statistical association*, ss. 366-372.