Answers to Review Comments

Vishal Balaji D and J. Arunnehru
Paper ID: 25

1. Which algorithm is used to fill the missing value in the dataset?

We fill in the missing attribute with the mode of that particular attribute in all the observations where the value of the target variable is the same as in the column with the missing value. The steps to do so are as follows:

1. For each observation with a missing value, the mode of the non-missing values of the variable with the missing value is calculated, for each value of the target variable. For example, in out dataset, the mode is found for all the available values of the **education** variable, for all the observations where the value of the target variable is *0*. The same is done for observations where the target variable is *1*.
2. The missing values are then filled with the calculated mode corresponding to the value of the target variable.
3. This process is repeated for each of the columns where values are missing.

2. Why one-hot encoding is used? Justify with your answer?

As mentioned in the dataset description, the target variable can have 2 values: *0*, which indicates that the particular employee has not been promoted or *1*, which indicates that the particular employee has been promoted.

The target variable is currently in integer encoding, which means that the possible values of this variable are in the form of a single integer, each representing one of the output classes. But, allowing the model to assume a natural order between these two categories may result in poor performance or unexpected results, such as predictions halfway between the categories. So, we encode the target variable using a method called One-hot encoding. One-hot encoding is a method of representing categorical variables in a more expressive manner. It helps to indicate to the algorithm that the expected output is to be categorical and not continuous.

To perform one-hot encoding, we use the integer representation of our target variable and transform them into an array of binary digits whose length is equal to the total number of possible values(2 in this case). The digit in the array whose position corresponds to our integer value is set to 1 while the other values are set to 0, i.e., *0* becomes *[1, 0]*, *1* becomes *[0, 1]* etc.

3. What is target variable in balancing class and what exactly 'yes' and 'no' represents here?

The variable to be predicted by the model, i.e., the target variable is the **is_promoted** column, which indicates whether the employee with particular attributes has been promoted or not. Our algorithms will be trained to output a prediction of what the value of this variable will be, based on the values of the other variables of a particular observation.

To aid in better visualization of the data, the values of the target variable *0* and *1* are represented as *no* and *yes* respectively in all figures, where no means that the particular employee has not been promoted and *yes* means that the employee has been promoted.

4. Why k-value was assigned to 5? Justify it with proper evidence and final value is 28.

To minimize computational power, considering that the dataset is fairly large in size, as well as to reduce the bias of the algorithm, we need to select a small value for K. Here, we arbitrarily choose K to be 5, since our dataset has 9330 observations, which can evenly be divided into 5 parts.

5. Kindly mention which attribute is used and why?

The statistics of the final trained models when evaluated on the testing set are tabulated below.

Table 1: Final statistics of the trained RF and XGB models evaluated on the testing set

Attribute	RF	XGB
Accuracy	0.8199	0.8339
P-Value [Acc > NIR]	< 2.2e-16	< 2.2e-16
Kappa	0.6399	0.6677
Sensitivity	0.7706	0.7835
Specificity	0.8692	0.8842
Balanced Accuracy	0.8199	0.8339

The above statistics are used to evaluate and describe the models. *Accuracy* is the percentage of correctly classified instances out of all instances. The *P-value* is a measure of the probability that an observed difference could have just by random chance. *Kappa* or *Cohen's Kappa* is similar to *Accuracy*, but it is normalized at the baseline of random chance on the dataset. *Sensitivity* describes the model's ability to predict true positives, while *Specificity* is a metric that evaluates the model's ability to predict true negatives. *Balanced Accuracy* is calculated as the average of the proportion of correct predictions from each individual class.

Note: The table presented above is modified when compared to the original manuscript. Some of the model statistics present in the original table were removed, since they proved to be irrelevant to describing these models in this context and therefore out of the scope of this paper.