CLUSTIMPUTE: DESCRIPTION OF THE ALGORITHM

OLIVER PFAFFEL

The intuition for the algorithm is that an observation should be clustered with other observations mainly based on their observed values (hence the weights on imputed values), while the resulting clusters provide donors for the missing value imputation, so that subsequently all variables can be used for clustering. It is usually recommended to standardize the data, if not measured on the same scale, prior to clustering so that each column has a mean of zero and a standard deviation of one. If all variables were measured on the same scale it is still important that they are centered before **ClustImpute** is applied with weights, i.e., with $n_{end} > 1$, otherwise the weighting procedure will introduce a bias. On a high-level the algorithm follows these steps:

1. Random imputation: replace all NAs by random imputation, i.e., for each variable with missings, draw from the marginal distribution of this variable excluding the missings. This does not take into account any correlations with other variables.
2. Weights < 1 are multiplied with imputed values to adjust their scale. The weights are calculated by a (linear) weight function that starts near zero and converges to 1 at n_{end}.
3. Regular k-means clustering with the Euclidean norm is performed with a number of c_{steps} steps starting with a random initialization.
4. The imputed values from step 2 are replaced by new draws conditionally on the cluster assignment from step 3.
5. Steps 2 to 4 are repeated nr_{iter} times in total. Any subsequent k-means clustering in step 3 uses the previous cluster centroids for initialization. Typically nr_{iter} is larger than n_{end}.
6. After the last draw of missing values a final k-means clustering is performed.

A very good overview on missing data imputation can be found in [2], for example. For k-means clustering we refer to the chapter on unsupervised learning in [1].

In the following we describe the clustering procedure formally and in more detail. We begin by describing classical k-means clustering and then highlight the difference of this implementation. First we describe the computation of the (hidden) k-th cluster centroid from all observations x assigned to cluster k. Let's assume there are N observations x in a p-dimensional space of real numbers and K clusters. Then each partition is characterized by a function $s : \{1, \dots, N\} \mapsto \{1, \dots, K\}$ that maps each observation to exactly one cluster. The computation of the k-th cluster centroid in the l-th iteration can be written as

$$c_k^l = \frac{1}{N_k^{l-1}} \sum_{\{i : s^{l-1}(i) = k\}} x_i \tag{1}$$

where N_k^{l-1} is the number of observations in cluster k, i.e., $N_k^{l-1} = |\{i : s^{l-1}(i) = k\}|$, and s^{l-1} is the partition function from the previous iteration. The initialization s^0 is typically random. The following step is to determine the closest centroid for each observation, i.e., to update the partition function:

$$s^l(i) = \arg\min_k \left\| x_i - c_k^l \right\|^2 \tag{2}$$

Here $\|\cdot\|$ denotes the Euclidean norm. In our setting $X = (x_{ij})$ has missing values. Therefore we are estimating

$$\tilde{X_{ij}}^{l} = \mathbb{1}_{\{x_{ij} \neq NA\}} x_{ij} + \mathbb{1}_{\{x_{ij} = NA\}} U_{ij}^l w(l), \tag{3}$$

(Dr. Oliver Pfaffel) Munich, Germany

where the weight function is given by $w(l) = \min\left(\frac{l}{n_{end}}, 1\right)$, and U_{ij}^l is uniformly distributed on all non-missing values of the same column j that lie in the same cluster $s^{l-1}(i)$, or all other variables if this set is empty. Mathematically, U_{ij}^l is uniformly distributed on

$$S_{ij}^l = \begin{cases} \{x_{rj} \neq NA : s^{l-1}(r) = s^{l-1}(i)\}, & \text{if non-empty} \\ \{x_{rj} \neq NA\}, & \text{otherwise.} \end{cases} \tag{4}$$

Thus the calculation of the new centroids is not only conditional on s but also on the realization of the random variable $U^l = (U_{ij}^l)$:

$$\tilde{c_k}^l = \frac{1}{N_k^{l-1}} \sum_{\{i:s^{l-1}(i)=k\}} \tilde{x_i}^l, \tag{5}$$

where U_{ij} is simply set to zero if x_{ij} is not missing. In early iterations, the weight function $w(l)$ is near zero, thus, for each component j, this is basically the mean over-all non missing values x_{ij}. Since the denominator N_k does not change with the share of missing values, there is some linear regularization towards zero, the mean due to standardization, proportional to the share of missing values. Finally, the update of the partition function,

$$s^l(i) = \arg\min_k \left(\left\| \tilde{x_i}^l - \tilde{c_k}^l \right\|^2 \right) \tag{6}$$

triggers, by definition, an update of S_{ij}, U and $\tilde{X}$.

References

[1] J. Friedman, T. Hastie, and R. Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

[2] S. Van Buuren. *Flexible imputation of missing data.* Chapman and Hall/CRC, 2018.