Interactions and Squares: Don't Transform, Just Impute!

Philipp Gaffert* Volker Bosch* Florian Meinfelder†

Abstract
Multiple imputation [Rubin, 1987] is difficult to conduct if the analysis model includes interactions, squares, or other transformations of variables with missing values for two reasons: First, the imputer must be aware of the analysis model to address the congeniality issue [Meng, 1994]. Second, the imputer must choose to produce either biased parameter estimates, even in the case of missing completely at random, by the passive-imputation algorithm (van Buuren and Groothuis-Oudshoorn [1999], a.k.a. impute, then transform) or inconsistent data relations by the just-another-variable algorithm (von Hippel [2009], a.k.a. transform, then impute). Although some research on imputing squares has been conducted [Vink and van Buuren, 2013], the conflict persists for all other non-trivial transformations. We propose a flexible local imputation model that builds upon the ideas of Cleveland [1979]. Implicitly, local imputation captures a broad range of transformations such as interactions, squares, cubes, roots, and logs. Hence, there is no need for the imputer to consider variable transformations. All they need to consider is the inclusion of all relevant variables as they are. In a simulation study, we compare our proposed local-imputation algorithm with, among others, random forest imputation [Doove et al., 2014], which also addresses nonlinearities.

Key Words: Multiple Imputation, Local Regression, Just Another Variable, Passive Imputation, Congeniality, Ignorability

1. Introduction

Multiple imputation has become one of the most popular methods for conducting statistical analysis with missing data. Its major advantage over alternatives, such as maximum likelihood [Little and Rubin, 2002], is that once the data are multiply imputed, the analyst can conduct any desired analysis. In contrast, maximum likelihood and other alternative methods have to be applied anew for each particular analysis. Multiple imputation originally involved drawing imputed values from multivariate posterior predictive distributions. These distributions are difficult to define in applications, which led to the development of fully conditional specification [Kennickell, 1991]. A detailed description of a fully-conditional-specification algorithm is given in van Buuren [2012].

In applications, however, where the analysis' objective is a regression model that contains interactions or other nonlinear transformations of the covariates, fully conditional multiple imputation yields biased and inconsistent results even if the missing mechanism is ignorable in the sense of Rubin [1976] [Seaman et al., 2012]. For the special case of a completely random missing mechanism, the just-another-variable algorithm developed by von Hippel [2009] provides consistent parameter estimation. For another special case in which logical consistency between the original variables and their transformations is key, the passive-imputation algorithm developed by van Buuren and Groothuis-Oudshoorn [1999] is suitable. Consequently, the current state of research is that, all of a sudden, the analysis objective does matter and affects the strategy to be applied within the multiple-imputation algorithm.

*Marketing & Data Sciences Division, GfK SE, Nordwestring 101, 90419 Nuremberg, Germany

†Department of Econometrics and Statistics, Otto-Friedrich-University, Feldkirchenstrasse 21, 96052 Bamberg, Germany

In this paper, we argue that, given the set of existing algorithms for imputing interactions, squares, and other nonlinear transformations, it is not worthwhile to build an imputation model that captures the functional relations well because algorithms that implicitly capture nonlinearities, such as the algorithm proposed by Doove et al. [2014], are superior. We further introduce a new algorithm of this type named local imputation and present its performance in a simulation study.

2. Imputing interactions, squares and other nonlinear transformations

The data consist of n realizations of a three-dimensional random vector (X, Y, Z). The bivariate distribution of (X, Z) is normal. y and x are fully observed. We refer to the n_{mis} and the $n_{obs} = n - n_{mis}$ realizations with and without missing values in z as recipients and donors, respectively. The missing mechanism is defined by $Pr\left(z = missing\right) = \Phi\left(x + \eta\right)$, with Φ denoting the normal cumulative distribution function and η denoting independent normal noise. The estimands are the parameters β of the regression model

$$Y = \beta_0 + \beta_x X + \beta_z Z + u + \sum_l \beta_l g_l\left(X, Z\right), \tag{1}$$

where u denotes independent normal noise and g is a functional relation. For simplicity, we assume that the model in equation (1) reflects both the true data generating process for Y and the analysis model. In the simplest case, in which $g\left(X, Z\right) = 0$, fully conditional parametric imputation proceeds as follows [Little and Rubin, 2002]:

1. On the fully observed part of the data, make a draw from the posterior distribution of the parameters $(\alpha, \sigma_\epsilon^2)$ of the imputation model $z^{obs} = \alpha_0 + \alpha_x x^{obs} + \alpha_y y^{obs} + \epsilon$.
2. Conditional on the drawn parameters $(\tilde{\alpha}, \tilde{\sigma}_\epsilon^2)$ and $\left(x^{mis}, y^{mis}\right)$, draw n_{mis} times independently from the imputation model to obtain imputations for z^{mis}.
3. Repeat the above steps $M \geqslant 2$ times, and then, apply Rubin's rules [Rubin, 1987].

Algorithm 1: Fully conditional normal imputation.

For more complex g, there are two approaches in the fully conditional setup: the passive-imputation algorithm [van Buuren and Groothuis-Oudshoorn, 1999] and the just-another-variable algorithm [von Hippel, 2009]. To introduce both approaches, suppose that $g\left(X, Z\right) = Z^2$. The data set then consists of four columns (y, x, z, z^2), where the first two columns are fully observed and the other two columns have missing values for exactly the same observations. The passive-imputation algorithm proceeds as algorithm 1 but includes the squared term in the imputation model for z as well. After imputation of z, it simply computes the squares. This is why von Hippel [2009] named this approach 'impute (the linear terms), then transform'. The just-another-variable algorithm imputes the missing values in z just like the passive-imputation algorithm. However, instead of just calculating the square from the imputed values, the algorithm repeats the imputation procedure for z^2 and thereby treats the transformation as if it were just another variable. Logical inconsistencies between z and its transformations g are a natural consequence of this procedure. If the missing pattern is completely at random [Mealli and Rubin, 2015], the just-another-variable algorithm enables consistent estimation of the parameters of interest, whereas the passive-imputation algorithm does not. For a missing at random pattern, neither of the two approaches provide consistent estimates [Seaman et al., 2012]. Vink and van Buuren [2013] proposed a solution for the special case of a squared term.

3. Omniscient versus ignorant imputers

Both the passive-imputation algorithm and the just-another-variable algorithm require the imputer to know the analysis model, which in our setup is equivalent to the data generating process. This seems a doable requirement, as all the imputer needs to do is talk to the analyst. However, in many applications, talking to the analyst is a tricky task. Let us consider public use files, where one imputer at the agency provides the file but where hundreds of analysts run highly sophisticated models driven by theories from their fields. Some very talented imputers out there might actually be capable of doing this job. However, there is no doubt that this takes substantial time and effort. Now, recall that if the missing data pattern is not missing completely at random, neither of the two existing approaches provide consistent parameter estimates; therefore, what is the reward for all this work? The new local-imputation algorithm that we propose in this paper does not require knowledge about g and can be shown to be consistent for a broad class of functional relations. We compare the local imputation with two other algorithms that are ignorant of the true g and also with the passive-imputation algorithm and the just-another-variable algorithm, which both utilize the true g and are thus referred to as omniscient.

4. Algorithms for ignorant imputers

Any given data set confronts the ignorant imputer with a classic bias-variance trade-off. A very sparse imputation model, which consists of for instance linear effects only, gives high-precision parameter estimates. However, any unconsidered relations are assumed to be zero. On the other hand, an imputation model with many parameters can be very inefficient if, e.g., the analysis model is sparse. Knowing the true model the omniscient imputer does not suffer from this uncertainty. Because the ignorant imputer is ignorant of the nonlinear transformations, logical inconsistencies between the variable with missing values and its transformations cannot occur, as opposed to the omniscient just-another-variable algorithm.

The motivation for the papers by Burgette and Reiter [2010] and Doove et al. [2014] is presumably highly similar to the motivation for our paper. Doove et al. [2014] propose to use random forests as implemented in Liaw and Wiener [2002] and show that the nature of the classification algorithm helps preserve interaction effects. Their approach is implemented in the R-package mice (R Core Team [2016], van Buuren and Groothuis-Oudshoorn [2011]). Doove et al. [2014] present coverages of frequentist confidence intervals to check whether the uncertainty associated with the estimation of the parameters of the imputation model is well propagated. However, because they choose a fairly large number of donors, the degree of uncertainty involved is low. Another potential issue is that most of the analysis models in applications involve some sort of regression, and thus, an imputation model based on classification inevitably causes uncongeniality [Meng, 1994].

As opposed to the random forest multiple-imputation algorithm by Doove et al. [2014], predictive mean matching as proposed by Little [1988] utilizes the parametric posterior step as in algorithm 1 but substitutes the imputation step with a nearest neighbor approach. Predictive mean matching is known to add robustness toward model misspecifications [Schenker and Taylor, 1996].

5. The proposed algorithm

5.1 Local regression

Local regression is proposed by Cleveland [1979] and Cleveland and Devlin [1988] to smooth scatterplots. Müller [1996] shows that local regression achieves a good fit under mild assumptions, namely, a sufficient number of observations in the neighborhood and continuity and differentiability of the underlying functional relation g. Without loss of generality algorithm 2 shows local regression in a setup with an $(n \times 2)$ predictor matrix $A = (x, y)$ and a response variable z to ensure consistent notation.

1. Divide the columns of A by their interquartile ranges to rescale, and add a constant column. Obtain A^*.
2. For any point a_0^*, select the neighborhood of the k nearest points with an observed z using Euclidean distances d_i.
3. For each observation in the neighborhood of a_0^*, excluding a_0^* itself, compute the weights w_i from the distances with the tricube function $w_i = \left[1 - \{d_i/max(d_i)\}^3\right]^3 + \delta$, with δ denoting a small positive scalar number.
4. Calculate $P = \left(A^{*\top}_{i=1\dots k}\right)\{diag(w_i)\}$, $Q = (P)\left(A^*_{i=1\dots k}\right)$, and Λ as the diagonal of Q, with the element corresponding to the constant in the model set to zero.
5. Estimate the parameters of a weighted regression model by $\hat{\gamma} = (Q + \lambda\Lambda)^{-1}(Pz)$.
6. Predict $\hat{z}_0 = a_0^*\hat{\gamma}$.

Algorithm 2: Local regression.

Especially for small k, dimensionality issues may occur [Marimont and Shapiro, 1979]. Choosing the scalar $\lambda > 0$ makes the weighted least squares regression a ridge regression with all its positive properties [Dempster et al., 1977] and thereby mixes constancy with linearity within the neighborhood [Cleveland and Devlin, 1988].

5.2 Distance-based donor selection

The posterior step of the distance-based donor selection algorithm [Siddique and Belin, 2008] consists of maximum likelihood estimation of the imputation model parameters on M independent bootstrap samples, thereby replacing the draws from the posterior distribution [Little and Rubin, 2002]. The imputations stem from drawing donors. For recipient j, donor i is drawn with probability

$$v_{i,j} = f(\omega, \hat{z}_i, \hat{z}_j, \kappa) = \omega_i(\Delta\hat{z})^{-\kappa}_{i,j} / \sum_{i=1}^{n_{obs}} \left\{\omega_i(\Delta\hat{z})^{-\kappa}_{i,j}\right\}. \tag{2}$$

The drawing probability $v_{i,j}$ depends upon the donor's bootstrap weight ω_i and the scalar absolute distance between the predictive means of donor i and recipient j, labeled as $(\Delta\hat{z})_{i,j}$. The closeness parameter κ controls the importance of the distance. Some improvements to this method are developed in an unpublished working paper available from the first author.

5.3 Local imputation

Now that we have collected all the ingredients for local multiple imputation, the algorithm is quite straightforward.

Algorithm 3 approaches the bias-variance trade-off by optimizing the number of parameters to estimate on a subsample of the data. One potential outcome is that $k = n_{nobs}$

1. Perform a Bayesian bootstrap and obtain the nonnegative bootstrap weights ω [Rubin, 1981]. Conduct all subsequent steps on the bootstrap sample.
2. To obtain optimal parameters for (k, λ), minimize the objective function $\|z - \hat{z}\|$, with $\hat{z}$ from algorithm 2 on a subsample of the donors.
3. For each observation, i.e., both donors and recipients, make a local regression prediction using the optimal choices for (k, λ) and algorithm 2.
4. Obtain the imputations by applying the distance-based donor selection algorithm as in equation (2), but draw only from the k donors inside the neighborhood.
5. Repeat the steps above for iterating over all variables and multiple imputations.

Algorithm 3: Local imputation.

so that the neighborhoods do not differ among recipients. This should occur for $g = 0$. If g is highly nonlinear and if the data are rather sparse, even a local regression line might not result in a good fit. The predictive-mean-matching-like distance-based donor selection algorithm can cure slight model misspecifications on the local level.

By using the bootstrap weights ω in both the posterior and the imputation step, it is ensured that the uncertainty involved in estimating the imputation model parameters is propagated.

Because local regression can fit any differentiable function, local imputation is a priori inclusive, i.e., the vast majority of potential relations g are not restricted. Typical analysis models are at least approximately, rather than literally, nested in the imputation model. Thus, estimates are likely to be consistent yet inefficient [Raghunathan, 2016]. Xie and Meng [2016] call this feature hidden robustness.

6. Simulation study

6.1 Simulation setup

Using a simulation study, we assess the relative and absolute performance of the proposed algorithm. We distinguish between ignorant approaches that take the linear terms as an input only and omniscient approaches that utilize the linear terms and the relevant transformations. The ignorant approaches are random forest imputation by Doove et al. [2014], the version of predictive mean matching in van Buuren [2012], and our proposed method; the omniscient approaches are the passive-imputation algorithm [van Buuren and Groothuis-Oudshoorn, 1999] and the just-another-variable algorithm [von Hippel, 2009]. The last two approaches also deploy predictive mean matching in the imputation step to achieve better comparability with the ignorant approaches.

We assume (X, Z) to follow a standard normal distribution with a $\rho = 0.2$ correlation. The missingness is always at random [Mealli and Rubin, 2015] and defined by $Pr\left(z = missing\right) = \Phi\left[(1/4)\{x + N(0,3)\}\right]$. We fix $M = 10$, $n_{mis} = 90$, and n_{obs} as low as 60 to obtain a substantial degree of estimation uncertainty of the imputation model parameters. Throughout the different analysis models, we keep the coefficient of determination at approximately $R^2 = 2/3$, and for each model, we perform $n_{sim} = 1000$ Monte Carlo simulation runs.

Table 1: Simulation results

	Ignorant approaches						Omniscient approaches			
	LI		PMM		RF		PI		JAV	
	R	C	R	C	R	C	R	C	R	C
Linear only*	$\sum_l \beta_l g_l(X,Z) = 0$									
$\beta_0 = 0$	126	(930)	119	(944)	146	(928)	120	(937)	118	(944)
$\beta_x = 1$	132	(938)	127	(953)	151	(925)	127	(949)	127	(948)
$\beta_z = 1$	131	(908)	120	(910)	150	(806)	121	(916)	122	(915)
Square	$\sum_l \beta_l g_l(X,Z) = \beta_{z2}Z^2$									
$\beta_0 = 0$	254	(945)	286	(907)	242	(898)	458	(346)	200	(948)
$\beta_x = 1$	249	(931)	267	(947)	235	(953)	440	(946)	195	(937)
$\beta_z = 1$	246	(952)	274	(919)	231	(937)	463	(951)	192	(923)
$\beta_{z2} = 1$	248	(930)	254	(846)	240	(819)	460	(178)	198	(911)
Interaction	$\sum_l \beta_l g_l(X,Z) = \beta_{xz}XZ$									
$\beta_0 = 0$	279	(939)	341	(962)	264	(934)	282	(871)	223	(889)
$\beta_x = 1$	280	(941)	348	(939)	265	(941)	279	(925)	218	(849)
$\beta_z = 1$	270	(808)	318	(717)	252	(789)	269	(833)	215	(860)
$\beta_{xz} = 1$	282	(791)	344	(537)	266	(720)	275	(459)	213	(881)
Cube	$\sum_l \beta_l g_l(X,Z) = \beta_{z3}Z^3$									
$\beta_0 = 0$	474	(949)	519	(957)	516	(957)	792	(982)	936	(961)
$\beta_x = 1$	461	(941)	494	(951)	503	(952)	755	(869)	949	(914)
$\beta_z = 0$	475	(963)	516	(924)	512	(954)	747	(913)	926	(717)
$\beta_{z3} = 1$	469	(964)	492	(951)	496	(907)	783	(943)	998	(669)

LI: Local imputation; PMM: Predictive mean matching; RF: Random forest imputation; PI: Passive imputation; JAV: Just another variable; R: Root Mean Squared Error $\times 1000$; C: Coverage of the 950‰ interval; *When $g = 0$, PMM, PI, and JAV are identical algorithms

6.2 Simulation results

Table 1 shows the results for all parameters of interest and all introduced imputation methods in two dimensions. The root mean squared error R measures how far on average the estimate on the imputed data set is away from the true value, be it due to bias or variance. Small values indicate good quality. The root mean squared error is a quality indicator in descriptive statistics. To obtain a quality indicator in inferential statistics, we construct $950‰$ confidence intervals using Rubin's rules (Rubin [1987], Barnard and Rubin [1999]). In each simulation run, we note whether this confidence interval covers the true parameter. Good quality is indicated by coverage values C of approximately $950‰$.

When averaging over all parameters, the proposed local-imputation algorithm obtains the best performance in terms of descriptive statistics, closely followed by random forest imputation. All other algorithms are far off. In terms of inferential statistics, local imputation comprehensively outperforms all alternatives. Here, random forest imputation also performs the second best; however, it is on average almost twice as far away from the theoretical $950‰$ benchmark. Regarding both evaluation criteria, the omniscient approaches perform the poorest despite having access to valuable additional information, namely, the true model.

Although local imputation achieves the best performance, the coverage values of approximately $800‰$ for the relation including the interaction term leave substantial room for improvement.

7. Conclusion

Whenever nonlinear relations are of interest, finding a suitable imputation model can turn out to be a serious burden for an imputer. Having established the model, the imputer can apply one of the two omniscient algorithms: the passive-imputation algorithm by van Buuren and Groothuis-Oudshoorn [1999] or the just-another-variable algorithm by von Hippel [2009]. We find that both of these algorithms, which incorporate information about the functional relation, perform poorly. This finding is in agreement with Seaman et al. [2012].

We refer to the algorithms that only use the linear terms and ignore the functional relation as ignorant. Ignorant algorithms are much easier to deploy, as there is no need to worry about the functional relation for the imputer. In this paper, we propose a new ignorant algorithm named local imputation. Its theoretical properties are well understood, as it is closely related to local regression by Cleveland [1979]. In the simulation study, we find that local imputation is superior to all other compared algorithms in terms of both descriptive and inferential statistics. Nevertheless, random forest imputation by Doove et al. [2014] performs almost equally well in terms of descriptive statistics.

We generally feel that there must be better ways to incorporate the knowledge of the true functional form. For the omniscient approaches, all the information necessary to perform a sensible imputation is available in our simulation study, but the results are disillusioning. This clearly indicates a lack of suitable algorithms. For random forest imputation, we believe that the uncertainty involved in parameter estimation must be propagated in a different manner to obtain correct inferences. An R-package providing the proposed local-imputation algorithm is available from the first author upon request [R Core Team, 2016].

Acknowledgements

We would like to thank Susanne Raessler and Trivellore Raghunathan for valuable thoughts and comments they gave on this research.

References

Barnard, J. and D. B. Rubin (1999). Small-sample degrees of freedom with multiple imputation. *Biometrika 86*(4), 948–955.

Burgette, L. F. and J. P. Reiter (2010). Multiple imputation for missing data via sequential regression trees. *American Journal of Epidemiology 172*(9), 1070–1076.

Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association 74*(368), 829–836.

Cleveland, W. S. and S. J. Devlin (1988). Locally weighted regression: an approach to regression analysis by local fitting. *Journal of the American Statistical Association 83*(403), 596–610.

Dempster, A. P., M. Schatzoff, and N. Wermuth (1977). A simulation study of alternatives to ordinary least squares. *Journal of the American Statistical Association 72*(357), 77–91.

Doove, L. L., S. van Buuren, and E. Dusseldorp (2014). Recursive partitioning for missing data imputation in the presence of interaction effects. *Computational Statistics & Data Analysis 72*, 92–104.

Kennickell, A. B. (1991). Imputation of the 1989 survey of consumer finances: Stochastic relaxation and multiple imputation. In *Proceedings of the Survey Research Methods Section of the American Statistical Association*, pp. 1–10.

Liaw, A. and M. Wiener (2002). Classification and regression by randomforest. *R News 2*(3), 18–22.

Little, R. J. (1988). Missing-data adjustments in large surveys (with discussion). *Journal of Business & Economic Statistics 6*(3), 287–301.

Little, R. J. and D. B. Rubin (2002). *Statistical Analysis with Missing Data 2nd ed.* New York: John Wiley & Sons.

Marimont, R. B. and M. B. Shapiro (1979). Nearest neighbour searches and the curse of dimensionality. *IMA Journal of Applied Mathematics 24*(1), 59–70.

Mealli, F. and D. B. Rubin (2015). Clarifying missing at random and related definitions, and implications when coupled with exchangeability. *Biometrika 102*(4), 995–1000.

Meng, X.-L. (1994). Multiple-imputation inferences with uncongenial sources of input. *Statistical Science 9*, 538–558.

Müller, W. G. (1996). Optimal design for local fitting. *Journal of Statistical Planning and Inference 55*(3), 389–397.

R Core Team (2016). *R 3.3.1: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

Raghunathan, T. (2016). *Missing Data Analysis in Practice*. Boca Raton: Chapman & Hall/CRC.

Rubin, D. B. (1976). Inference and missing data. *Biometrika 63*(3), 581–592.

Rubin, D. B. (1981). The bayesian bootstrap. *The Annals of Statistics 9*(1), 130–134.

Rubin, D. B. (1987). *Multiple Imputation for Nonresponse in Surveys*. New York: John Wiley & Sons.

Schenker, N. and J. M. Taylor (1996). Partially parametric techniques for multiple imputation. *Computational Statistics & Data Analysis 22*(4), 425–446.

Seaman, S. R., J. W. Bartlett, and I. R. White (2012). Multiple imputation of missing covariates with non-linear effects and interactions: an evaluation of statistical methods. *BMC medical research methodology 12*(1), 1–13.

Siddique, J. and T. R. Belin (2008). Multiple imputation using an iterative hot-deck with distance-based donor selection. *Statistics in Medicine 27*(1), 83–102.

van Buuren, S. (2012). *Flexible Imputation of Missing Data*. Boca Raton: Chapman & Hall/CRC.

van Buuren, S. and K. Groothuis-Oudshoorn (1999). Flexible multivariate imputation by mice. Technical report, Leiden, The Netherlands: TNO Prevention and Health.

van Buuren, S. and K. Groothuis-Oudshoorn (2011). mice: Multivariate imputation by chained equations in R. *Journal of Statistical Software 45*(3), 1–67.

Vink, G. and S. van Buuren (2013). Multiple imputation of squared terms. *Sociological Methods & Research 42*(4), 598–607.

von Hippel, P. T. (2009). How to impute interactions, squares, and other transformed variables. *Sociological Methodology 39*(1), 265–291.

Xie, X. and X.-L. Meng (2016). Dissecting multiple imputation from a multi-phase inference perspective: What happens when gods, imputers and analysts models are uncongenial? *Statistica Sinica Preprint*, 1–56.